D.D. KERR
VE3FGK

MODUCOMP INC.

75 CALIFORNIA AVENUE, BROCKVILLE, ONTARIO, CANADA K6V 5Y6
(613-342-5041)

MOD 80 SUPPLEMENT

PRINTED IN CANADA

CONTENTS

1. 8080 Component Description

The entire 8080 microprocessor prototyping system may be upgraded to use the more powerful 8080 CPU, thus allowing logic system designers to implement a prototyping system to evaluate the 8080 for their specific application.

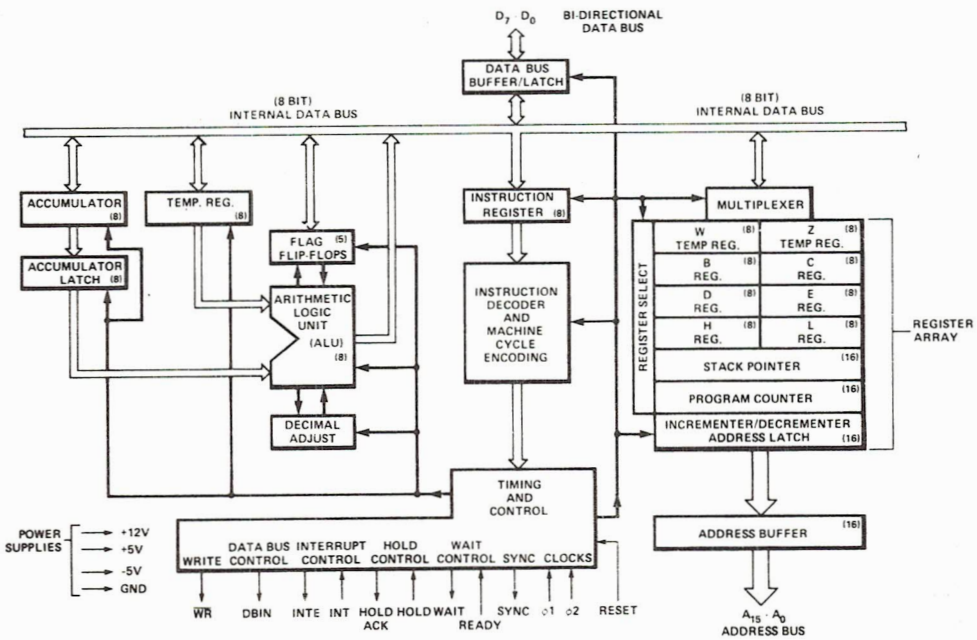
The 8080 is a monolithic 8 bit central processing unit for use in general purpose digital computer systems. It is fabricated using N-channel Si gate process technology. The 8080 contains six 8 bit data registers, an 8 bit accumulator, four 8 bit temporary registers, four testable flag bits and 8 bit parallel binary and decimal arithmetic units. The 8080 also provides 16 bit arithmetic and immediate operations, which greatly simplify memory address calculations and high speed arithmetic operations. The 8080 has a 16 bit program counter, which allows direct addressing of up to 64K bytes of memory. The 8080 also contains a 16 bit stack pointer to control the addressing of an external last-in/first-out stack. The stack can be used to store and restore complete system status thus facilitating the handling of multi-level interrupts. Another major advantage of the stack is that it allows virtually unlimited subroutine nesting. Although significantly higher in performance than other existing microprocessors, the 8080 has been designed to simplify system design. Separate 16 bit address and 8 bit data busses are used to allow direct interfacing to memory and I/O parts. A HOLD signal provides the ability to suspend processor operation and force the address and data bus drivers into a high-impedance state. This simplifies the implementation of direct memory access and multi-processor systems.

1.1 Absolute Maximum Ratings

Temperature Under Bias	0° C to +70° C	V_{CC}, V_{DD} and V_{SS} With	
Storage Temperature	−65° C to +150° C	Respect to V_{BB}	−0.3V to +20V
All Input or Output Voltages		Power Dissipation	1.5W
With Respect to V_{BB}	−0.3V to +20V		

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

1.2 Block Diagram



1.3 Pin Configuration

8080 Functional Pin Definition
The following describes the function of all the 8080 I/O pins. Several of the descriptions refer to internal timing periods.

$A_{15} - A_0$ (Output Three-State)
ADDRESS BUS; the address bus provides the memory address or denotes the I/O number for up to 256 input and 256 output devices.

$D_7 - D_0$ (Input/Output Three-State)
DATA BUS; the data bus provides bi-directional communication between the CPU, memory, and I/O devices for instructions and data transfers.

SYNC (Output)
SYNCHRONIZING SIGNAL; the SYNC pin provides a signal at the beginning of each machine cycle.

DBIN (Output)
DATA BUS IN; the DBIN signal indicates that the data bus is in the input mode. This signal should be used to enable the gating of data onto the 8080 data bus from memory or I/O.

READY (Input)
READY; the READY signal indicates to the 8080 that valid memory or input data is available on the data bus. This signal is used to synchronize the CPU with slower memory or I/O devices. After sending an address out the 8080 will enter a WAIT state for as long as the READY line is low.

WAIT (Output)
WAIT; the WAIT signal acknowledges that the CPU is in a WAIT state.

WR (Output)
WRITE; the WR signal is used for memory WRITE or I/O output control. The data on the data bus is stable while the WR signal is low.

HOLD (Input)
HOLD; the HOLD signal requests the CPU to enter the HOLD state. This state allows an external device to gain control of the 8080 address and data bus as soon as the 8080 has completed its use of these buses for the current machine cycle. It is recognized under the following conditions:

- the CPU is in the HALT state.
- the CPU is in the T2 or TW state and the READY signal is active.

As a result of entering the HOLD state the CPU ADDRESS BUS ($A_{15} - A_0$) and DATA BUS ($D_7 - D_0$) will be in their high impedance state.

HLDA (Output)
HOLD ACKNOWLEDGE; the HLDA signal appears in response to the HOLD signal and indicates that the data and address bus will go to the high impedance state. The HLDA signal begins at:

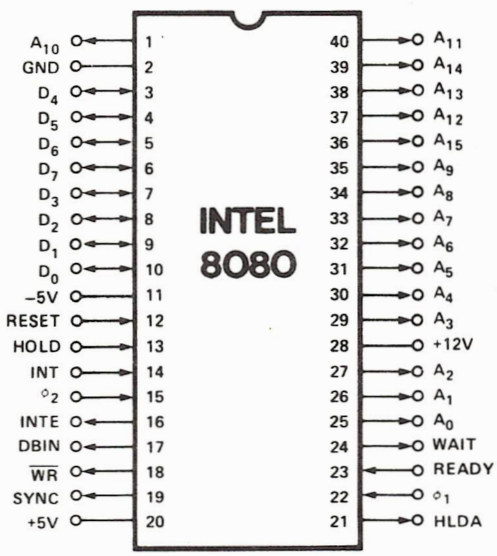
- T3 for READ memory or input.
- The Clock Period following T3 for WRITE memory or OUTPUT operation.

In either case, the HLDA signal appears after the rising edge of $\phi 1$ and high impedance occurs after the rising edge of $\phi 2$.

INTE (Output)
INTERRUPT ENABLE; indicates the content of the internal interrupt enable flip/flop may be set or reset by the Enable and Disable Interrupt instructions. It inhibits interrupts from being accepted by the CPU when it is reset. It is automatically reset at time T1 of the instruction fetch cycle (M1) when an interrupt is accepted. It is also reset by the RESET signal.

INT (Input)
INTERRUPT REQUEST; the CPU recognizes an interrupt reque on this line at the end of the current instruction or while halted. If the CPU is in the HOLD state or if the Interrupt Enable flip/flop is reset it will not honor the request.

RESET (Input)
RESET; while the RESET signal is activated, the content of the program counter is cleared and the instruction register is set to 0. After RESET, the program counter is cleared and the instruction register is set to 0. After RESET, the program will start at location 0 in memory. The INTE and HLDA flip/flops are also reset. Note that the flags, accumulator, and registers are not cleared. Reset must be valid for at least 3 clock cycles.



PIN NO.	PIN DESIGNATION	PIN NO.	PIN DESIGNATION	PIN NO.	PIN DESIGNATION	PIN NO.	PIN DESIGNATION
1	ADDRESS 10	11	-5V	21	HLDA	31	ADDRESS 5
2	GND	12	RESET	22	$\phi 1$	32	ADDRESS 6
3	DATA IN/OUT 4	13	HOLD	23	READY	33	ADDRESS 7
4	DATA IN/OUT 5	14	INT	24	WAIT	34	ADDRESS 8
5	DATA IN/OUT 6	15	$\phi 2$	25	ADDRESS 0	35	ADDRESS 9
6	DATA IN/OUT 7	16	INTE	26	ADDRESS 1	36	ADDRESS 15
7	DATA IN/OUT 3	17	DBIN	27	ADDRESS 2	37	ADDRESS 12
8	DATA IN/OUT 2	18	$\overline{WR}$	28	+12V	38	ADDRESS 13
9	DATA IN/OUT 1	19	SYNC	29	ADDRESS 3	39	ADDRESS 14
10	DATA IN/OUT 0	20	+5V	30	ADDRESS 4	40	ADDRESS 11

1.4 Electrical Characteristics

T_A = 0°C, to 70°C, V_{DD} = +12V ± 5%, V_{CC} = +5V ± 5%, V_{BB} = -5V ± 5%, V_{SS} = 0V, Unless Otherwise Noted.

Symbol	Parameter	Min.	Typ.	Max.	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	$V_{SS}-1$		$V_{SS}+0.6$	V	
V_{IHC}	Clock Input High Voltage	$V_{DD}-1$		$V_{DD}+1$	V	
V_{IL}	Input Low Voltage	$V_{SS}-1$		$V_{SS}+0.8$	V	
V_{IH}	Input High Voltage	3.3		$V_{CC}+1$	V	
V_{OL}	Output Low Voltage			0.45	V	I_{OL} = 1.7mA on the Data Bus I_{OL} = .75mA on all other outputs
V_{OH}	Output High Voltage	3.7			V	I_{OH} = 100μA.
$I_{DD\,(AV)}$	Avg. Power Supply Current (V_{DD})		40	67	mA	Operation
$I_{CC\,(AV)}$	Avg. Power Supply Current (V_{CC})		60	75	mA	T_A = 25°C
$I_{BB\,(AV)}$	Avg. Power Supply Current (V_{BB})		.01	1	mA	T_{CY} = .48 μsec
I_{IL}	Input Leakage			±10	μA	$V_{SS} \leqslant V_{IN} \leqslant V_{CC}$
I_{CL}	Clock Leakage			±10	μA	$V_{SS} \leqslant V_{CLOCK} \leqslant V_{DD}$
I_{DL} [3]	Data Bus Leakage in Input Mode			-100	μA	$V_{SS} \leqslant V_{IN} \leqslant V_{CC}$
I_{FL}	Address and Data Bus Leakage During HOLD			+10 -100	μA	$V_{ADDR/DATA} = V_{CC}$ $V_{ADDR/DATA} = V_{SS}$

1.5 Capacitance

T_A = 25°C $V_{CC} = V_{DD} = V_{SS}$ = 0V, V_{SS} = -5V ±5%

Symbol	Parameter	Typ.	Max.	Unit	Test Condition
C_ϕ	Clock Capacitance	10	20	pf	f_c = 1 MHz
C_{IN}	Input Capacitance	5	10	pf	Unmeasured Pins
C_{OUT}	Output Capacitance	10	20	pf	Returned to V_{SS}

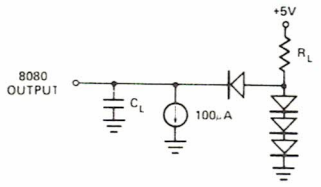
1.6 AC Characteristics

T_A = 0°C to 70°C, V_{DD} = +12V ± 5%, V_{CC} = +5V ± 5%, V_{BB} = -5V ± 5%, V_{SS} = 0V, Unless Otherwise Noted

Symbol	Parameter	Min.	Max.	Unit	Test Condition
t_{CY} [3]	Clock Period	0.48	2.0	μsec	
t_r, t_f	Clock Rise and Fall Time	5	50	nsec	
$t_{\phi 1}$	ϕ_1 Pulse Width	60		nsec	
$t_{\phi 2}$	ϕ_2 Pulse Width	220		nsec	
t_{D1}	Delay ϕ_1 to ϕ_2	0		nsec	
t_{D2}	Delay ϕ_2 to ϕ_1	70		nsec	
t_{D3}	Delay ϕ_1 to ϕ_2 Leading Edges	130		nsec	
t_{DA} [2]	Address Output Delay From ϕ_2		200	nsec	R_L = 4.5kΩ, C_L = 100pf
t_{DD} [2]	Data Output Delay From ϕ_2		220	nsec	R_L = 2.1kΩ, C_L = 100pf
t_{DC} [2]	Signal Output Delay From ϕ_1, or ϕ_2 (SYNC, $\overline{WR}$ WAIT HLDA)		120	nsec	R_L = 4.5kΩ, C_L = 50pf
t_{DF} [2]	DBIN Delay From ϕ_2	25	140	nsec	R_L = 2.1kΩ, C_L = 50pf
t_{DI} [1]	Delay for Input Bus to Enter Input Mode During DBIN		t_{DF}	nsec	
t_{DS1}	Data "Setup Time" During ϕ_1 and DBIN	50		nsec	
t_{DS2}	Data "Setup Time" to ϕ_2 During DBIN	150		nsec	
t_{DH} [1]	Data "Hold Time" From ϕ_2 During DBIN	t_{DF}		nsec	
t_{IE} [2]	INTE Output Delay From ϕ_2		200	nsec	R_L = 4.5kΩ, C_L = 50pf
t_{RS}	Ready "Setup Time" During ϕ_2	120		nsec	
t_{HS}	Hold "Setup Time" to ϕ_2	140		nsec	
t_{IS}	INT "Setup Time" During ϕ_2 (During ϕ_1 in Halt Mode)	180		nsec	
t_H	"Hold Time" From ϕ_2 (Ready, INT, Hold)	0		nsec	
t_{FD}	Delay to Float During Hold (Address and DATA BUS)		120	nsec	
t_{WA} [2]	Address Stable From $\overline{WR}$	t_{D3}		nsec	R_L = 4.5kΩ, C_L = 100pf
t_{AW} [2]	Address Stable Prior to $\overline{WR}$	[5]		nsec	R_L = 4.5kΩ, C_L = 100pf

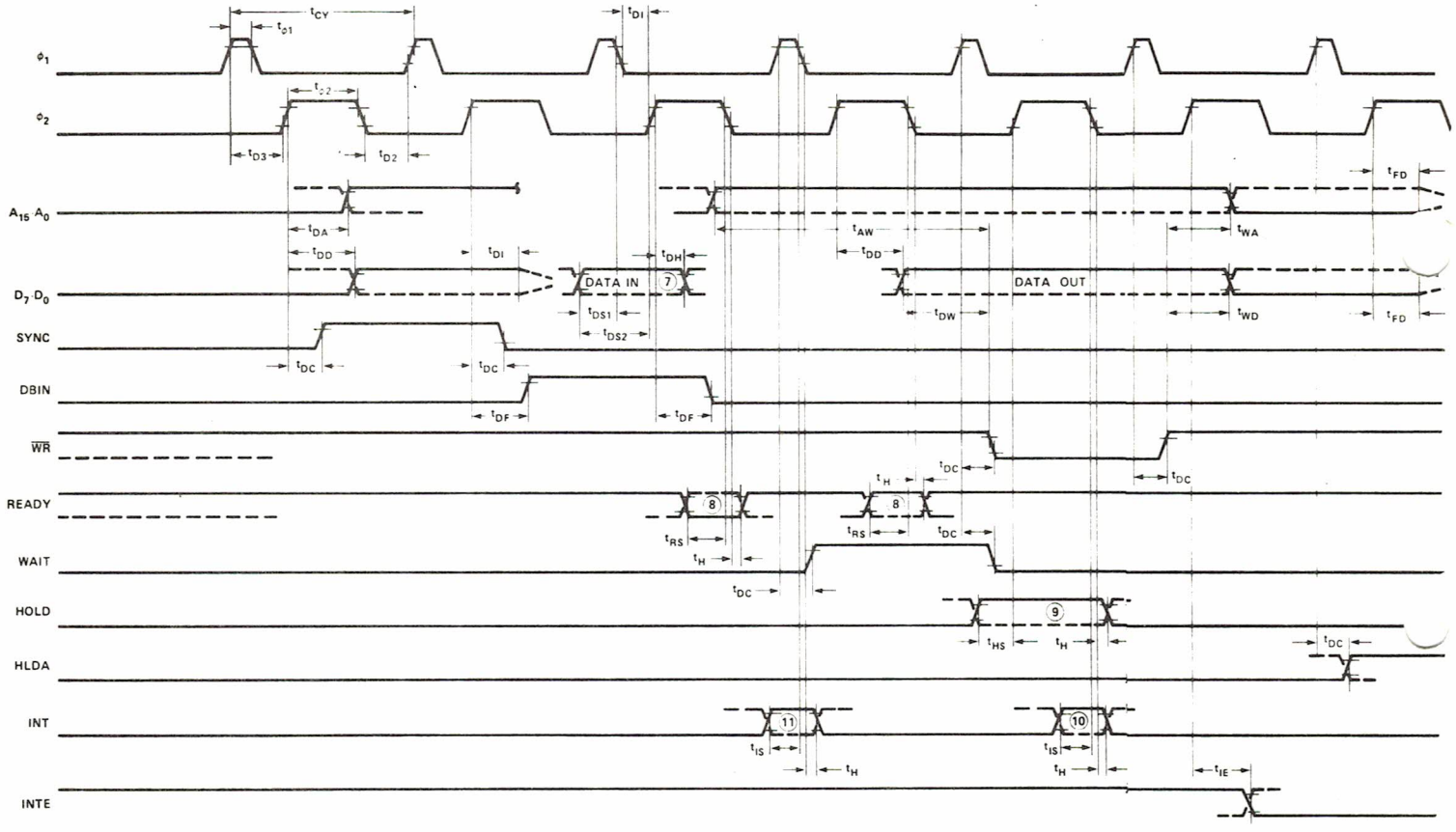
Symbol	Parameter	Min.	Max.	Unit	Test Condition
t_{WD}[2]	Output Data Stable From $\overline{WR}$	t_{D3}		nsec	R_L = 2.1kΩ, C_L = 100pf
t_{DW}[2]	Output Data Stable Prior to $\overline{WR}$	[6]		nsec	R_L = 2.1kΩ, C_L = 100pf

NOTES:
1. Data input should be enabled with DBIN status. No bus conflict can then occur and data hold time is assured.
2. Load circuit
3. $t_{CY} = t_{D3} + t_{r\phi2} + t_{\phi2} + t_{f\phi2} + t_{D2} + t_{r\phi1} \geq 480$ns.
4. The following are relevant when interfacing the 8080 to devices having $V_{IH} = 3.3V$:
   a) Maximum output rise time from .8V to 3.3V = 140ns @ C_L = SPEC.
   b) Output delay when measured to 3.0V = SPEC + 60ns @ C_L = SPEC.
   c) If $C_L \neq$ SPEC add .6ns/pf if $C_L > C_{SPEC}$, subtract .3ns/pf (from modified delay) if $C_L < C_{SPEC}$.
5. $t_{AW} = 2\,t_{CY} - t_{D3} - t_{r\phi2} - 120$nsec.
6. $t_{DW} = t_{CY} - t_{D3} - t_{r\phi2} - 150$nsec.
7. Data in must be stable for this period during DBIN * T_3. Both t_{DS1} and t_{DS2} must be satisfied.
8. Ready signal must be stable for this period during T_2 or T_W. (Must be externally synchronized.)
9. Hold signal must be stable for this period during T_2 or T_W when entering hold mode, and during T_3, T_4, T_5 and T_{WH} when in hold mode. (Must be externally synchronized.)
10. Interrupt signal must be stable during this period of the last clock cycle of any instruction to be recognized on the following instruction. (External synchronization is not required.)
11. During halt mode only, timing is with respect to ϕ_1 falling edge.
12. This timing diagram shows timing relationships only, it does not represent any specific machine cycle.



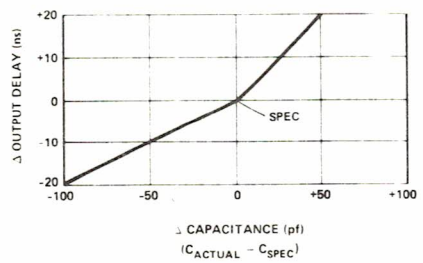
1.7 Switching Characteristics

(Note: Timing measurements are made at the following reference voltages: CLOCK "1" = 9.5V, "0" = 1.0V; INPUTS "1" = 3.3V, "0" = 0.8V; OUTPUTS "1" = 2.0V, "0" = 0.8V.)

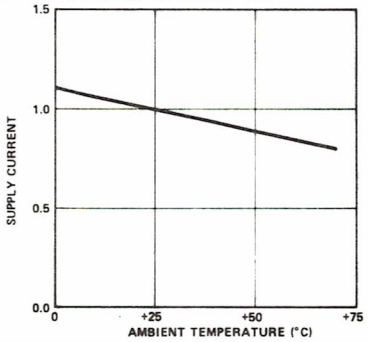


1.8 Typical Characteristic Curves

TYPICAL Δ OUTPUT DELAY VS. Δ CAPACITANCE



TYPICAL SUPPLY CURRENT VS. TEMPERATURE, NORMALIZED.



1.9 Instruction Set

The 8080 has many instructions which are extremely useful and extend the range of applicability of the CPU. The main instruction groups are as follows:

- Data register and memory transfers.
- Conditional and unconditional branches and calls.
- Input/output operations
- Direct load and store of accumulator
- Save and restore data registers, accumulator and flags.
- Double length operations in data registers
  - Increment/Decrement/Add
  - Direct load and store (H and L)
  - Load immediate
  - Index register modification/exchange
- Indirect jump
- Stack pointer modification
- Logical operations
- Binary arithmetic
- Decimal arithmetic
- Set and reset interrupt enable flip-flop
- Increment/decrement memory or data registers
- Special Instructions
  - NO-OP (No operation)
  - HALT (Stop the CPU)

Data and Instruction Formats

The program instructions may be one, two, or three bytes in length. Multiple byte instructions must be stored in successive words in program memory. The instruction formats then depend on the particular operation executed.

One Byte Instructions
D7 D6 D5 D4 D3 D2 D1 D0 OP CODE

Two Byte Instructions
D7 D6 D5 D4 D3 D2 D1 D0 OP CODE
D7 D6 D5 D4 D3 D2 D1 D0 OPERAND

Three Byte Instructions
D7 D6 D5 D4 D3 D2 D1 D0 OP CODE
D7 D6 D5 D4 D3 D2 D1 D0 LOW ADDRESS OR OPERAND 1
D7 D6 D5 D4 D3 D2 D1 D0 HIGH ADDRESS OR OPERAND 2

For the 8080 a logic "1" is defined as a high level and a logic "0" is defined as a low level.

1.10 Summary of Processor Instructions

Mnemonic	Description	Instruction Code[1] D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Clock[2] Cycles
MOV$_{r1, r2}$	Move register to register	0	1	D	D	D	S	S	S	5
MOV M, r	Move register to memory	0	1	1	1	0	S	S	S	7
MOV r, M	Move memory to register	0	1	D	D	D	1	1	0	7
HLT	Halt	0	1	1	1	0	1	1	0	7
MVI r	Move immediate register	0	0	D	D	D	1	1	0	7
MVI M	Move immediate memory	0	0	1	1	0	1	1	0	10
INR r	Increment register	0	0	D	D	D	1	0	0	5
DCR r	Decrement register	0	0	D	D	D	1	0	1	5
INR M	Increment memory	0	0	1	1	0	1	0	0	10
DCR M	Decrement memory	0	0	1	1	0	1	0	1	10
ADD r	Add register to A	1	0	0	0	0	S	S	S	4
ADC r	Add register to A with carry	1	0	0	0	1	S	S	S	4
SUB r	Subtract register from A	1	0	0	1	0	S	S	S	4
SBB r	Subtract register from A with borrow	1	0	0	1	1	S	S	S	4
ANA r	And register with A	1	0	1	0	0	S	S	S	4
XRA r	Exclusive Or register with A	1	0	1	0	1	S	S	S	4
ORA r	Or register with A	1	0	1	1	0	S	S	S	4
CMP r	Compare register with A	1	0	1	1	1	S	S	S	4
ADD M	Add memory to A	1	0	0	0	0	1	1	0	7
ADC M	Add memory to A with carry	1	0	0	0	1	1	1	0	7
SUB M	Subtract memory from A	1	0	0	1	0	1	1	0	7
SBB M	Subtract memory from A with borrow	1	0	0	1	1	1	1	0	7
ANA M	And memory with A	1	0	1	0	0	1	1	0	7
XRA M	Exclusive Or memory with A	1	0	1	0	1	1	1	0	7
ORA M	Or memory with A	1	0	1	1	0	1	1	0	7
CMP M	Compare memory with A	1	0	1	1	1	1	1	0	7
ADI	Add immediate to A	1	1	0	0	0	1	1	0	7
ACI	Add immediate to A with carry	1	1	0	0	1	1	1	0	7
SUI	Subtract immediate from A	1	1	0	1	0	1	1	0	7
SBI	Subtract immediate from A with borrow	1	1	0	1	1	1	1	0	7
ANI	And immediate with A	1	1	1	0	0	1	1	0	7
XRI	Exclusive Or immediate with A	1	1	1	0	1	1	1	0	7
ORI	Or immediate with A	1	1	1	1	0	1	1	0	7
CPI	Compare immediate with A	1	1	1	1	1	1	1	0	7
RLC	Rotate A left	0	0	0	0	0	1	1	1	4
RRC	Rotate A right	0	0	0	0	1	1	1	1	4
RAL	Rotate A left through carry	0	0	0	1	0	1	1	1	4
RAR	Rotate A right through carry	0	0	0	1	1	1	1	1	4
JMP	Jump unconditional	1	1	0	0	0	0	1	1	10
JC	Jump on carry	1	1	0	1	1	0	1	0	10
JNC	Jump on no carry	1	1	0	1	0	0	1	0	10
JZ	Jump on zero	1	1	0	0	1	0	1	0	10
JNZ	Jump on no zero	1	1	0	0	0	0	1	0	10
JP	Jump on positive	1	1	1	1	0	0	1	0	10
JM	Jump on minus	1	1	1	1	1	0	1	0	10
JPE	Jump on parity even	1	1	1	0	1	0	1	0	10
JPO	Jump on parity odd	1	1	1	0	0	0	1	0	10

Mnemonic	Description	Instruction Code[1] D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Clock[2] Cycles
CALL	Call unconditional	1	1	0	0	1	1	0	1	17
CC	Call on carry	1	1	0	1	1	1	0	0	11/17
CNC	Call on no carry	1	1	0	1	0	1	0	0	11/17
CZ	Call on zero	1	1	0	0	1	1	0	0	11/17
CNZ	Call on no zero	1	1	0	0	0	1	0	0	11/17
CP	Call on positive	1	1	1	1	0	1	0	0	11/17
CM	Call on minus	1	1	1	1	1	1	0	0	11/17
CPE	Call on parity even	1	1	1	0	1	1	0	0	11/17
CPO	Call on parity odd	1	1	1	0	0	1	0	0	11/17
RET	Return	1	1	0	0	1	0	0	1	10
RC	Return on carry	1	1	0	1	1	0	0	0	5/11
RNC	Return on no carry	1	1	0	1	0	0	0	0	5/11
RZ	Return on zero	1	1	0	0	1	0	0	0	5/11
RNZ	Return on no zero	1	1	0	0	0	0	0	0	5/11
RP	Return on positive	1	1	1	1	0	0	0	0	5/11
RM	Return on minus	1	1	1	1	1	0	0	0	5/11
RPE	Return on parity even	1	1	1	0	1	0	0	0	5/11
RPO	Return on parity odd	1	1	1	0	0	0	0	0	5/11
RST	Restart	1	1	A	A	A	1	1	1	11
IN	Input	1	1	0	1	1	0	1	1	10
OUT	Output	1	1	0	1	0	0	1	1	10
LXI B	Load immediate register Pair B & C	0	0	0	0	0	0	0	1	10
LXI D	Load immediate register Pair D & E	0	0	0	1	0	0	0	1	10
LXI H	Load immediate register Pair H & L	0	0	1	0	0	0	0	1	10
LXI SP	Load immediate stack pointer	0	0	1	1	0	0	0	1	10
PUSH B	Push register Pair B & C on stack	1	1	0	0	0	1	0	1	11
PUSH D	Push register Pair D & E on stack	1	1	0	1	0	1	0	1	11
PUSH H	Push register Pair H & L on stack	1	1	1	0	0	1	0	1	11
PUSH PSW	Push A and Flags on stack	1	1	1	1	0	1	0	1	11
POP B	Pop register pair B & C off stack	1	1	0	0	0	0	0	1	10
POP D	Pop register pair D & E off stack	1	1	0	1	0	0	0	1	10
POP H	Pop register pair H & L off stack	1	1	1	0	0	0	0	1	10
POP PSW	Pop A and Flags off stack	1	1	1	1	0	0	0	1	10
STA	Store A direct	0	0	1	1	0	0	1	0	13
LDA	Load A direct	0	0	1	1	1	0	1	0	13
XCHG	Exchange D & E, H & L Registers	1	1	1	0	1	0	1	1	4
XTHL	Exchange top of stack, H & L	1	1	1	0	0	0	1	1	18
SPHL	H & L to stack pointer	1	1	1	1	1	0	0	1	5
PCHL	H & L to program counter	1	1	1	0	1	0	0	1	5
DAD B	Add B & C to H & L	0	0	0	0	1	0	0	1	10

DAD D	Add D & E to H & L	0	0	0	1	1	0	0	1	10
DAD H	Add H & L to H & L	0	0	1	0	1	0	0	1	10
DAD SP	Add stack pointer to H & L	0	0	1	1	1	0	0	1	10
STAX B	Store A indirect	0	0	0	0	0	0	1	0	7
STAX D	Store A indirect	0	0	0	1	0	0	1	0	7
LDAX B	Load A indirect	0	0	0	0	1	0	1	0	7
LDAX D	Load A indirect	0	0	0	1	1	0	1	0	7
INX B	Increment B & C registers	0	0	0	0	0	0	1	1	5
INX D	Increment D & E registers	0	0	0	1	0	0	1	1	5
INX H	Increment H & L registers	0	0	1	0	0	0	1	1	5
INX SP	Increment stack pointer	0	0	1	1	0	0	1	1	5

DCX B	Decrement B & C	0	0	0	0	1	0	1	1	5
DCX D	Decrement D & E	0	0	0	1	1	0	1	1	5
DCX H	Decrement H & L	0	0	1	0	1	0	1	1	5
DCX SP	Decrement stack pointer	0	0	1	1	1	0	1	1	5
CMA	Compliment A	0	0	1	0	1	1	1	1	4
STC	Set carry	0	0	1	1	0	1	1	1	4
CMC	Compliment carry	0	0	1	1	1	1	1	1	4
DAA	Decimal adjust A	0	0	1	0	0	1	1	1	4
SHLD	Store H & L direct	0	0	1	0	0	0	1	0	16
LHLD	Load H & L direct	0	0	1	0	1	0	1	0	16
EI	Enable Interrupts	1	1	1	1	1	0	1	1	4
DI	Disable interrupt	1	1	1	1	0	0	1	1	4
NOP	No-operation	0	0	0	0	0	0	0	0	4

NOTES: 1. DDD or SSS – 000 B – 001 C – 010 D – 011 E – 100 H – 101 L – 110 Memory – 111 A.
2. Two possible cycle times (5/11) indicate instruction cycles dependent on condition flags.

1.11 Status Information

Instructions for the 8080 require from one to five machine cycles for complete execution. The 8080 sends out 8 bit of status information on the data bus at the beginning of each machine cycle (during SYNC time). The following table defines the status information.

Status Information Definition

Symbols	Data Bus Bit	Definition
HLTA	D_3	Acknowledge signal for HALT instruction.
INTA*	D_0	Acknowledge signal for INTERRUPT request. Signal should be used to gate a restart instruction onto the data bus when DBIN is active.
INP*	D_6	Indicates that the address bus contains the address of an input device and the input data should be placed on the data bus when DBIN is active.
OUT	D_4	Indicates that the address bus contains the address of an output device and the data bus will contain the output data when $\overline{WR}$ is active.
MEMR*	D_7	Designates that the data bus will be used for memory read data.
M_1	D_5	Provides a signal to indicate that the CPU is in the fetch cycle for the first byte of an instruction.
STACK	D_2	Indicates that the address bus holds the pushdown stack address from the Stack Pointer.
$\overline{WO}$	D_1	Indicates that the operation in the current machine cycle will be a WRITE memory or OUTPUT function ($\overline{WO} = 0$). Otherwise, a READ memory or INPUT operation will be executed.

*These three status bits can be used to control the flow of data onto the 8080 data bus.

1.12 Timing

Instructions in the 8080 contain one to three bytes. Each instruction requires from one to five machine or memory cycles for fetching and execution. Machine cycles are called M1, M2, . . . , M5. Each machine cycle requires from three to five states T1, T2, . . . , T5 for its completion. Each state has the duration of one clock period (0.5 micro-second). There are three other states (WAIT, HOLD, and HALT) which last one to an indefinite number of clock periods, as controlled by external signals. Machine cycle M1 is always the operation-code fetch cycle and lasts four or five clock periods. Machine cycles M2, M3, M4, and M5 normally last three clock periods each.

To understand the basic operation of the 8080, refer to the simplified state diagram and the timing diagram.

During T1 the content of the program counter is sent to the address bus, SYNC is true, and the data bus contains the status information pertaining to the cycle that is currently being initiated. T1 is always followed by another state, T2, during which the condition of the READY, HOLD and HALT Acknowledge Signals are tested. If READY is true, T3 can be entered; otherwise, the CPU will go into the wait state (TW) and stay there for as long as READY is false.

READY thus allows the CPU speed to be synchronized to a memory with any access time or to any input device. Furthermore, by properly controlling the READY line, the user can single-step through his program.

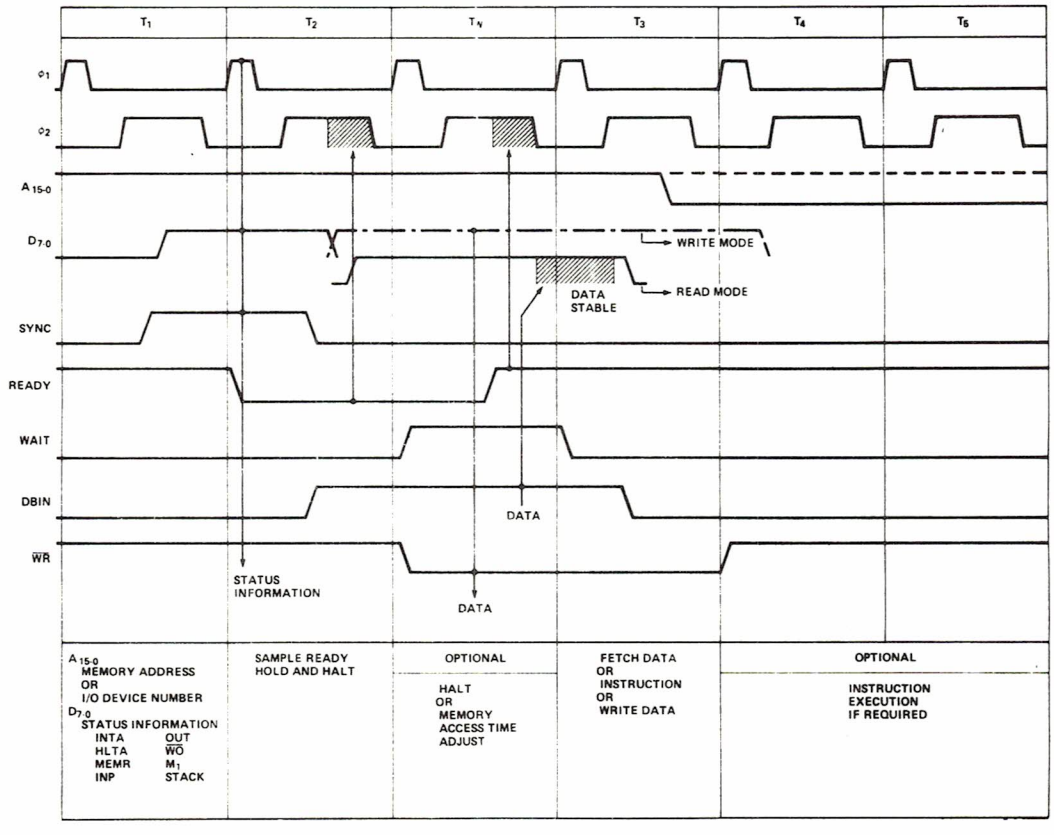
During T3, the data coming from memory is available on the data bus and is transferred into the instruction register (during M1 only) as shown in the 8080 block diagram of Figure 34. The instruction decoder and control sections then generate the basic signals to control the internal data transfers, the timing, and the machine cycle requirements of the new instructions.

At the end of T4, if the cycle is complete, or else at the end of T5, the 8080 goes back to T1 and enters machine cycle M2, unless the instruction required only one machine cycle for its execution. In such cases, a new M1 cycle is entered. The loop is repeated for as many cycles and states as required by the instruction.

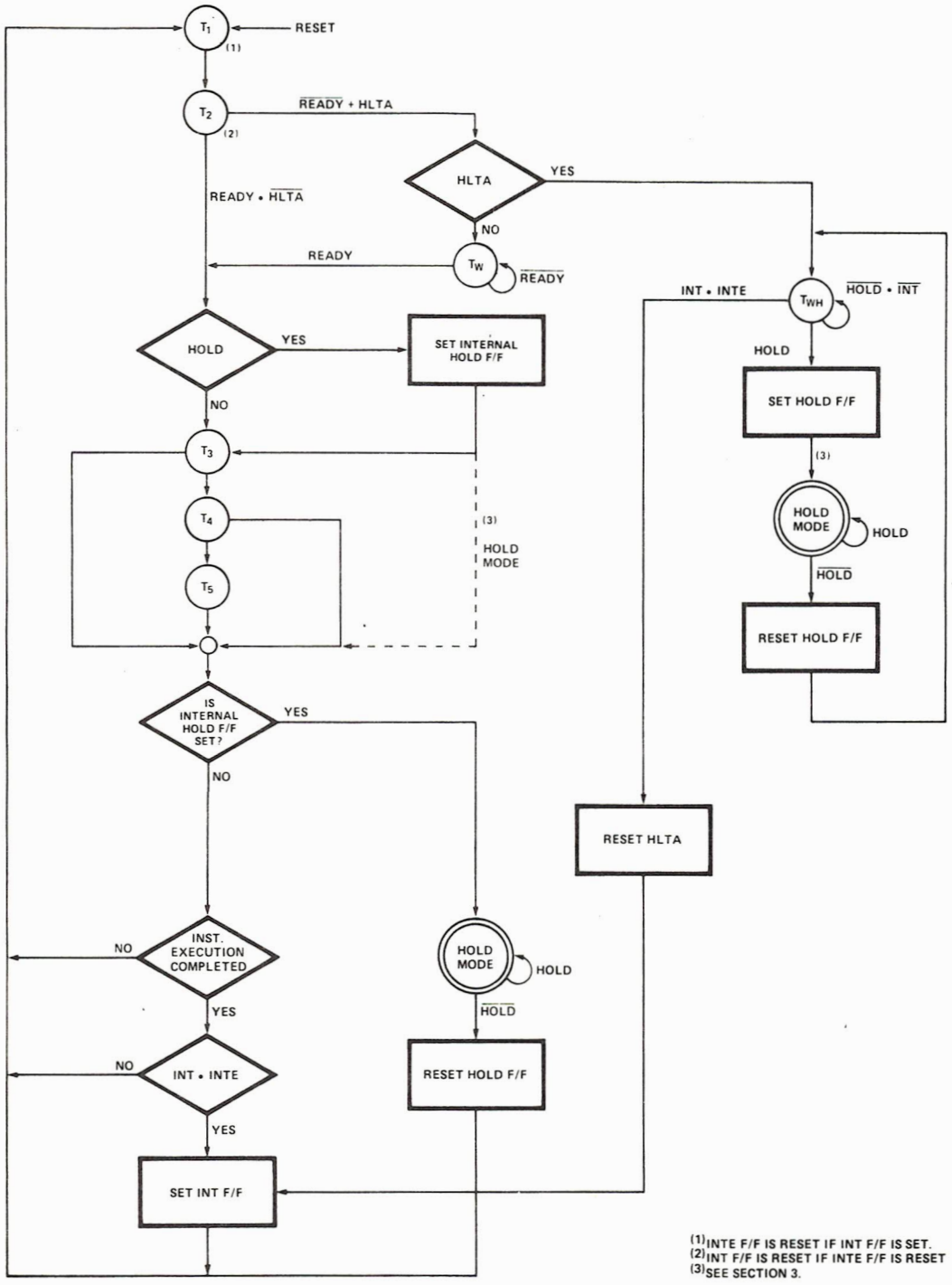
It is only during the last state of the last machine cycle that the interrupt request line is tested and a special M1 cycle is entered, during which no program-counter incrementing takes place and INTERRUPT ACKNOWLEDGE status is sent out. During this cycle, one of eight possible restart instructions will be sent to the CPU by the interrupting device.

Instruction state requirements range from a minimum of four states for non-memory referencing instructions, like register and accumulator arithmetic instructions, up to a maximum of 18 states for the most complex instructions (exchange the contents of registers H and L with the content of the top two locations of the stack). At the maximum clock frequency of 2 megahertz, this means that all instructions will be executed in intervals ranging from 2 μs to 9μs. If a HALT instruction is executed, the processor enters a WAIT state and remains there until an interrupt is received.

Basic 8080 Instruction Cycle



CPU State Transition Diagram



(1) INTE F/F IS RESET IF INT F/F IS SET.
(2) INT F/F IS RESET IF INTE F/F IS RESET
(3) SEE SECTION 3.

2 Processor Programming Instructions

2.1 Complete Functional Definition

The following pages present a detailed description of the complete 8080 Instruction Set.

Symbols	Meaning
<B2>	Second byte of the instruction
<B3>	Third byte of the instruction
r	One of the scratch pad register references: A, B, C, D, E, H, L
c	One of the following flag flip-flop references:

flag flip-flops.

Condition for True	
carry	— Overflow, underflow
zero	— Result is zero
sign	— MSB of result is "1"
parity	— Parity of result is even

M	Memory location indicated by the contents of registers H and L
( )	Contents of location or register
Λ	Logical product
∀	Exclusive "or"
V	Inclusive "or"
r_m	Bit m of register r
SP	Stack Pointer
PC	Program Counter
←	Is transferred to
XXX	A "don't care"
SSS	Source register for data
DDD	Destination register for data

Register # (SSS or DDD)	Register Name
000	B
001	C
010	D
011	E
100	H
101	L
110	Memory
111	ACC

8080 INSTRUCTION SET

Mnemonic	Bytes	Cycles	Description of Operation
MOV r_1, r_2	1	1	$(r_1) \leftarrow (r_2)$ Load register r_1 with the content of r_2. The content of r_2 remains unchanged.
MOV r, M	1	2	(r) ← (M) Load register r with the content of the memory location addressed by the contents of registers H and L.
MOV M, r	1	2	(M) ← (r) Load the memory location addressed by the contents of registers H and L with the content of register r.
MVI r <B_2>	2	2	(r) ← <B_2> Load byte two of the instruction into register r.
MVI M <B_2>	2	3	(M) ← <B_2> Load byte two of the instruction into the memory location addressed by the contents of registers H and L.
INR r	1	1	(r) ← (r) + 1 The content of register r is incremented by one. All the condition flip-flops except carry are affected by the result.
DCR r	1	1	(r) ← (r) − 1 The content of register r is decremented by one. All of the condition flip-flops except carry are affected by the result.
ADD r	1	1	(A) ← (A) + (r) Add the content of register r to the content of register A and place the result into register A. (All flags affected.)
ADC r	1	1	(A) ← (A) + (r) + (carry) Add the content of register r and the contents of the carry flip-flop to the content of the A register and place the result into Register A. (All flags affected.)
SUB r	1	1	(A) ← (A) − (r) Subtract the content of register r from the content of register A and place the result into register A. Two's complement subtraction is used. (All flags affected.)
SBB r	1	1	(A) ← (A) − (r) − (borrow) Subtract the content of register r and the content of the carry flip-flop from the content of register A and place the result into register A. (All flags affected.)
ANA r	1	1	(A) ← (A) Λ (r) Place the logical product of the register A and register r into register A. (Resets carry.)
XRA r	1	1	(A) ← (A) ∀ (r) Place the "exclusive - or" of the content of register A and register r into register A. (Resets carry.)
ORA r	1	1	(A) ← (A) V (r) Place the "inclusive - or" of the content of register A and register r into register A. (Resets carry.)
CMP r	1	1	(A) − (r) Compare the content of register A with the content of register r. The content of register A remains unchanged. The flag flip-flops are set by the result of the subtraction. Equality (A = r) is indicated by the zero flip-flop set to "1." Less than (A < r) is indicated by the carry flip-flop, set to "1."
ADD M	1	2	(A) ← (A) + (M) ADD
ADC M	1	2	(A) ← (A) + (M) + (carry) ADD with carry
SUB M	1	2	(A) ← (A) − (M) SUBTRACT
SBB M	1	2	(A) ← (A) − (M) − (borrow) SUBTRACT with borrow
ANA M	1	2	(A) ← (A) Λ (M) Logical AND
XRA M	1	2	(A) ← (A) ∀ (M) Exclusive OR
ORA M	1	2	(A) ← (A) V (M) Inclusive OR
CMP M	1	2	(A) − (M) COMPARE
ADI <B_2>	2	2	(A) ← (A) + <B_2> ADD
ACI <B_2>	2	2	(A) ← (A) + <B_2> + (carry) ADD with carry

(M) addressed by the contents of registers H and L.

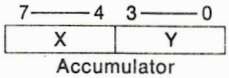
Flags affected are same as non-memory reference instructions.

Mnemonic	Bytes	Cycles	Description of Operation
SUI <B_2>	2	2	(A) ← (A) − <B_2> SUBTRACT
SBI <B_2>	2	2	(A) ← (A) − <B_2> − (borrow) SUBTRACT with borrow
ANI <B_2>	2	2	(A) ← (A) ∧ <B_2> Logical AND
XRI <B_2>	2	2	(A) ← (A) <B_2> Exclusive OR
ORI <B_2>	2	2	(A) ← (A) <B_2> Inclusive OR
CPI <B_2>	2	2	(A) − <B_2> COMPARE
RLC	1	1	$A_{m+1} \leftarrow A_m$, $A_0 \leftarrow A_7$, (carry) ← A_7 Rotate the content of register A left one bit. Rotate A_7 into A_0 and into the carry flip-flop.
RRC	1	1	$A_m \leftarrow A_{m+1}$, $A_7 \leftarrow A_0$, (carry) ← A_0 Rotate the content of register A right one bit. Rotate A_0 into A_7 and into the carry flip-flop.
RAL	1	1	$A_{m+1} \leftarrow A_m$, $A_0 \leftarrow$ (carry), (carry) ← A_7 Rotate the content of Register A left one bit. Rotate the content of the carry flip-flop into A_0. Rotate A_7 into the carry flip-flop.
RAR	1	1	$A_m \leftarrow A_{m+1}$, $A_7 \leftarrow$ (carry), (carry) ← A_0 Rotate the content of register A right one bit. Rotate the content of the carry flip-flop into A_7. Rotate A_0 into the carry flip-flop.
JMP <B_2> <B_3>	3	3	(PC) ← <B_3> <B_2> Jump unconditionally to the instruction located in memory location addressed by byte two and byte three.
JC <B_2> <B_3>	3	3	If (Carry) = 1 (PC) ← <B_3> <B_2> Otherwise (PC) = (PC) + 3
JNC <B_2> <B_3>	3	3	If (Carry) = 0 (PC ← <B_3> <B_2> Otherwise (PC) = (PC) + 3
JZ <B_2> <B_3>	3	3	If (Zero) = 1 (PC) ← <B_3> <B_2> Otherwise (PC) = (PC) + 3
JNZ <B_2> <B_3>	3	3	If (Zero) = 0 (PC) ← <B_3> <B_2> Otherwise (PC) = (PC) + 3
JP <B_2> <B_3>	3	3	If (Sign) = 0 (PC) ← <B_3> <B_2> Otherwise (PC) = (PC) + 3
JM <B_2> <B_3>	3	3	If (Sign) = 1 (PC) ← <B_3> <B_2> Otherwise (PC) = (PC) + 3
JPE <B_2> <B_3>	3	3	If (Parity) = 1 (PC) ← <B_3> <B_2> Otherwise (PC) = (PC) + 3
JPO <B_2> <B_3>	3	3	If (Parity) = 0 (PC) ← <B_3> <B_2> Otherwise (PC) = (PC) + 3
HLT	1	1	On receipt of the Halt Instruction, the activity of the processor is immediately suspended in the STOPPED state. The content of all registers and memory is unchanged and the PC has been updated.
CALL <B_2> <B_3>	3	5	[SP − 1] [SP − 2] ← (PC), (SP) = (SP) − 2 (PC) ← <B_3> <B_2> Transfer the content of PC to the pushdown stack in memory addressed by the register SP. The content of SP is decremented by two. Jump unconditionally to the instruction located in memory location addressed by byte two and byte three of the instruction.
CC <B_2> <B_3>	3	3/5	If (carry) = 1 [SP − 1] [SP − 2] ← PC, (SP) = (SP) − 2, (PC) ← <B_3> <B_2>; otherwise (PC) = (PC) + 3
CNC <B_2> <B_3>	3	3/5	If (carry) = 0 [SP − 1] [SP − 2] ← PC, (SP) = (SP) − 2, (PC) ← <B_3> <B_2>; otherwise (PC) = (PC) + 3
CZ <B_2> <B_3>	3	3/5	If (zero) = 1 [SP − 1] [SP − 2] ← PC, (SP) = (SP) − 2, (PC) ← <B_3> <B_2>; otherwise (PC) = (PC) + 3

Mnemonic	Bytes	Cycles	Description of Operation
CNZ <B_2> <B_3>	3	3/5	If (zero) = 0 [SP − 1] [SP − 2] ← PC, (SP) = (SP) − 2, (PC) ← <B_3> <B_2>; otherwise (PC) = (PC) + 3
CP <B_2> <B_3>	3	3/5	If (sign) = 0 [SP − 1] [SP − 2] ← PC, (SP) = (SP) − 2, (PC) ← <B_3> <B_2>; otherwise (PC) = (PC) + 3
CM <B_2> <B_3>	3	3/5	If (sign) = 1 [SP − 1] [SP − 2] ← PC, (SP) = (SP) − 2, (PC) ← <B_3> <B_2>; otherwise (PC) = (PC) + 3
CPE <B_2> <B_3>	3	3/5	If (parity) = 1 [SP − 1] [SP − 2] ← PC, (SP) = (SP) − 2, (PC) ← <B_3> <B_2>; otherwise (PC) = (PC) + 3
CPO <B_2> <B_3>	3	3/5	If (parity) = 0 [SP − 1] [SP − 2] ← PC, (SP) = (SP) − 2, (PC) ← <B_3> <B_2>; otherwise (PC) = (PC) + 3
RET	1	3	(PC) ← [SP] [SP + 1] (SP) = (SP) + 2. Return to the instruction in the memory location addressed by the last values shifted into the pushdown stack addressed by SP. The content of SP is incremented by two.
RC	1	1/3	If (carry) = 1 (PC) ← [SP], [SP + 1], (SP) = (SP) + 2; otherwise (PC) = (PC) + 1
RNC	1	1/3	If (carry) = 0 (PC) ← [SP], [SP + 1], (SP) = (SP) + 2; otherwise (PC) = (PC) + 1
RZ	1	1/3	If (zero) = 1 (PC) ← [SP], [SP + 1], (SP) = (SP) + 2; otherwise (PC) = (PC) + 1
RNZ	1	1/3	If (zero) = 0 (PC) ← [SP], [SP + 1], (SP) = (SP) + 2; otherwise (PC) = (PC) + 1
RP	1	1/3	If (sign) = 0 (PC) ← [SP], [SP + 1], (SP) = (SP) + 2; otherwise (PC) = (PC) + 1
RM	1	1/3	If (sign) = 1 (PC) ← [SP], [SP + 1], (SP) = (SP) + 2; otherwise (PC) = (PC) + 1
RPE	1	1/3	If (parity) = 1 (PC) ← [SP], [SP + 1], (SP) = (SP) + 2; otherwise (PC) = (PC) + 1
RPO	1	1/3	If (parity) = 0 (PC) ← [SP], [SP + 1], (SP) = (SP) + 2; otherwise (PC) = (PC) + 1
RST	1	3	[SP − 1] [SP − 2] ← (PC), (SP) = (SP) − 2 (PC) ← (00000000 00AAA000)
IN <B_2>	2	3	(A) ← (Input data) At T_1 time of third cycle, byte two of the instruction, which denotes the I/O device number, is sent to the I/O device through the address lines*, and the INP status information, instead of MEMR, is sent out at sync time. New data for the accumulator is loaded from the data bus when DBIN control signal is active. The condition flip-flops are not affected.
OUT <B_2>	2	3	(Output data) ← (A) At T_1 time of the third cycle, byte two of the instruction, which denotes the I/O device number, is sent to the I/O device through the address lines*, and the OUT status information is sent out at sync time. The content of the accumulator is made available on the data bus when the $\overline{WR}$ control signal is 0.
LXI B <B_2> <B_3>	3	3	(C) ← <B_2>; (B) ← <B_3> Load byte two of the instruction into C. Load byte three of the instruction into B.
LXI D <B_2> <B_3>	3	3	(E) ← <B_2>, (D) ← <B_3> Load byte two of the instruction into E. Load byte 3 of the instruction into D.
LXI H <B_2> <B_3>	3	3	(L) ← <B_2>, (H) ← <B_3> Load byte two of the instruction into L. Load byte three of the instruction into H.
*The device address appears on $A_7 - A_0$ and $A_{15} - A_8$			
LXI SP <B_2> <B_3>	3	3	$(SP)_L$ ← <B_2>, $(SP)_H$ ← <B_3> Load byte two of the instruction into the lower order 8-bit of the stack pointer and byte three into the higher order 8-bit of the stack pointer.

Mnemonic	Bytes	Cycles	Description of Operation
PUSH PSW	1	3	[SP − 1] ← (A), [SP − 2] ← (F), (SP) = (SP) − 2 Save the contents of A and F (5-flags) into the pushdown stack addressed by the SP register. The content of SP is decremented by two. The flag word will appear as follows: D_0: CY_2 (Carry) D_1: 1 D_2: Parity (even) D_3: 0 D_4: CY_1 D_5: 0 D_6: Zero D_7: MSB (sign)
PUSH B	1	3	[SP − 1] ← (B) [SP − 2] ← (C), (SP) = (SP) − 2
PUSH D	1	3	[SP − 1] ← (D) [SP − 2] ← (E), (SP) = (SP) − 2
PUSH H	1	3	[SP − 1] ← (H) [SP − 2] ← (L), (SP) = (SP) − 2
POP PSW	1	3	(F) ← [SP], (A) ← [SP + 1], (SP) = (SP) + 2 Restore the last values in the pushdown stack addressed by SP into A and F. The content of SP is incremented by two.
POP B	1	3	(C) ← [SP], (B) ← [SP + 1], (SP) = (SP) + 2
POP D	1	3	(E) ← [SP], (D) ← [SP + 1], (SP) = (SP) + 2
POP H	1	3	(L) ← [SP], (H) ← [SP + 1], (SP) = (SP) + 2
STA $<B_2>$ $<B_3>$	3	4	[$<B_3>$ $<B_2>$] ← (A) Store the accumulator content into the memory location addressed by byte two and byte three of the instruction.
LDA $<B_2>$ $<B_3>$	3	4	(A) ← [$<B_3>$ $<B_2>$] Load the accumulator with the content of the memory location addressed by byte two and byte three of the instruction.
XCHG	1	1	(H) ←→ (D) (E) ←→ (L) Exchange the contents of registers H and L and registers D and E.
XTHL	1	5	(L) ←→ [SP], (H) ←→ [SP + 1] Exchange the contents of registers H, L and the last values in the pushdown stack addressed by registers SP. The SP register itself is not changed. (SP) = (SP)
SPHL	1	1	(SP) ← (H) (L) Transfer the contents of registers H and L into register SP.
PCHL	1	1	(PC) ← (H) (L) JUMP INDIRECT
DAD SP	1	3	(H) (L) ← (H) (L) + (SP) Add the content of register SP to the content of registers H and L and place the result into registers H and L. If the overflow is generated, the carry flip-flop is set; otherwise, the carry flip-flop is reset. The other condition flip-flops are not affected. This is useful for addressing data in the stack.
DAD B	1	3	(H) (L) ← (H) (L) + (B) (C)
DAD H	1	3	(H) (L) ← (H) (L) + (H) (L) (double precision shift left H and L)
DAD D	1	3	(H) (L) ← (H) (L) + (D) (E)
STAX B	1	2	[(B) (C)] ← (A) Store the accumulator content in the memory location addressed by the content of registers B and C.
STAX D	1	2	[(D) (E)] ← (A) Store the accumulator content into the memory location addressed by the content of register D and E.
LDAX B	1	2	(A) ← [(B) (C)] Load the accumulator with the content of the memory location addressed by the content of registers B and C.
LDAX D	1	2	(A) ← [(D) (E)] Load the accumulator with the content of memory location addressed by the content of register D and E.
INX B	1	1	(B) (C) ← (B) (C) + 1 The content of register pair B and C is incremented by one. All of the condition flip-flops are not affected.
INX H	1	1	(H) (L) ← (H) (L) + 1 The content of register H and L is incremented by one. All of the condition flip-flops are not affected.
INX D	1	1	(D) (E) ← (D) (E) + 1
INX SP	1	1	(SP) ← (SP) + 1
DCX B	1	1	(B) (C) ← (B) (C) − 1
DCX H	1	1	(H) (L) ← (H) (L) − 1
DCX D	1	1	(D) (E) ← (D) (E) − 1
DCX SP	1	1	(SP) ← (SP) − 1
CMA	1	1	(A) ← $\overline{(A)}$ The content of accumulator is complemented. The condition flip-flops are not affected.

Mnemonic	Bytes	Cycles	Description of Operation
STC	1	1	(Carry) ← 1 Set the carry flip-flop to 1. The other condition flip-flops are not affected.
CMC	1	1	(carry) ← $\overline{(carry)}$ The content of carry is complemented. The other condition flip-flops are not affected.

DAA	1	1	Decimal Adjust Accumulator
			The 8-bit value in the accumulator containing the result from an arithmetic operation on decimal operands is adjusted to contain two valid BCD digits by adding a value according to the following rules:
			 
			
			Accumulator
			If (Y ≥ 10) or (carry from bit 3) then Y = Y + 6 with carry to X digit.
			If (X ≥ 10) or (carry from bit 7) or [(Y ≥ 10) and (X = 9)] then X = X + 6 (which sets the carry flip-flop).
			Two carry flip-flops are used for this instruction. CY_1 represents the carry from bit 3 (the fourth bit) and is accessible as a fifth flag. CY_2 is the carry from bit 7 and is the usual carry bit.
			All condition flip-flops are affected by this instruction.

SHLD <B_2> <B_3>	3	5	[<B_3> <B_2>] ← (L), [<B_3> <B_2> + 1] ← (H) Store the contents of registers H and L into the memory location addressed by byte two and byte three of the instructions.
LHLD <B_2> <B_3>	3	5	(L) ← [<B_3> <B_2>], (H) ← [<B_3> <B_2> + 1] Load the registers H and L with the contents of the memory location addressed by byte two and byte three of the instruction.
EI	1	1	Interrupt System Enable
DI	1	1	Interrupt System Disable The Interrupt Enable flip-flop (INTE) can be set or reset by using the above mentioned instructions. The INT signal will be accepted if the INTE is set. When the INT signal is accepted by the CPU, the INTE will be reset immediately. During interrupt enable or disable instruction executions, an interrupt will not be accepted.
INR M	1	3	[M] ← [M] + 1. The content of memory designated by registers H and L is incremented by one. All of the condition flip-flops except carry are affected by the result.
DCR M	1	3	[M] ← [M] −1. The content of memory designated by registers H and L is decremented by one. All of the condition flip-flops except carry are affected by the result.

2. MOD 80, an 8080 Based Modular Micro Computer

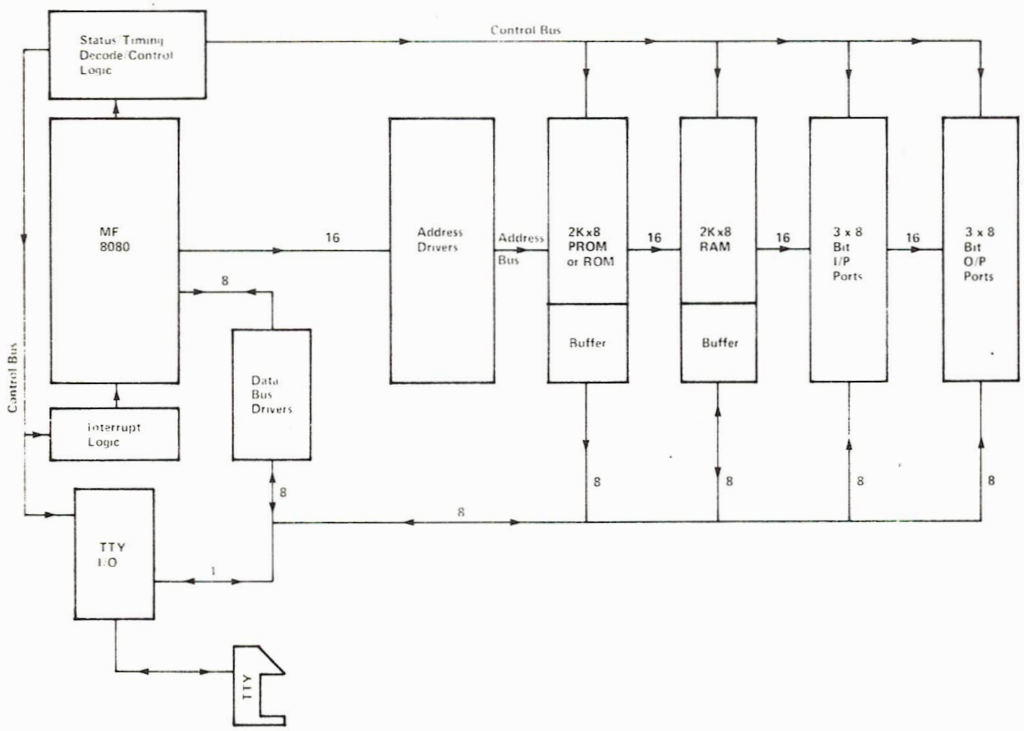
The MOD 80 is a modular 8080 based prototyping system. This hardware in conjunction with the MONITOR 80 software described in section 3 allows the user to establish his own hardware/software system to cater to his particular requirements.

The MOD80 uses the same backplane, TTY I/O, ROM and RAM boards as the MOD8 system.
The 8080 CPU board MOD80-1 plugs into the same backplane slot as was previously occupied by the MOD8-3 buffer board. The buffering of the 8080 address and data lines is accomplished on the MOD80-1 CPU board. The slot previously occupied by the MOD8-1 CPU board is now loaded with a simple jumper board. The function of this jumper board is to taylor the backplane to the 8080 CPU rather than the 8008 CPU.

The MOD80 system is available from:

MODUCOMP INC.
75 CALIFORNIA AVENUE,
BROCKVILLE,
ONTARIO,
CANADA K6V5Y6.

MOD80 System Configuration



2.1 MOD 80-1 Detailed CPU Board Description

The 8080 microprocessor (11) on this board requires 2 phase non-overlapping clocks. The clocks are generated using 2x74123 monostables. 15A and 15B are cascaded with Q of 15B connected back to A of 15A. This forms an oscillator with a period equal to the sum of the delays of 15A and 15B. The Q output of 15B drives the positive edge trigger input of 14A and the negative edge trigger input of 14B. The $\overline{Q}$ output of 14A provides a high level Ø1 pulse to the CPU via a 7416 open collector inverter with a 360Ω pull up to +12V. Similarly Ø2 is derived from the $\overline{Q}$ output of 14B. The delay of 14A determines the Ø1 pulse width and the delay of 14B the Ø2 pulse width. The system clock rate is memory limited to an effective CPU clock rate of 1 MHZ.

Ø1 width 120ns + tr @ 15ns and tf @ 15ns	= 150ns
Ø1 Ø2 interval	= 120ns
Ø2 width 300ns + tr @ 15ns and tf @ 15ns	= 330ns
Ø2 to Ø1 interval	= 400ns
Total = Clock cycle time	= 1 μs.

INTA and INTB are the inputs from a biased push button switch. INTA is normally grounded, but when the reset button is pressed INTB is grounded instead. This causes the cross coupled inverters (13) to toggle and the input to the CPU RESET terminal to switch from low to high for the duration that the push button remains depressed. Thus, pushing the reset button applies a debounced reset pulse to the CPU, causing it to start execution at location zero.

The SYNC and Ø1TTL are combined in a 2 input NAND gate (5) inverted (9) and applied to the enable pins of a 74100 eight bit latch (12). The inputs to 12 are DB0 thru DB7. At the start of each machine cycle the 8080 outputs a SYNC signal. When the SYNC signal and Ø1 are both high the information on the data bus is status information from the CPU defining what type of machine cycle is to be initiated.

Although all the status bits are latched, only the following are used to derive control signals on this card.

INTA	Acknowledge signal for INTERRUPT request.
OUT	Indicates the machine cycle will output information to an output port.
INP	Indicates the machine cycle will input information from an input port.
MEMR	Indicates that the data bus will be used for memory read data.

MEMR is combined with the buffered (9) DBIN signal from the CPU (DBIN high indicates that the required data bus flow is into the CPU.) The NAND combination (5) of DBIN and MEMR is inverted (4) to provide system control signal MRE, which is valid or high when the CPU wishes to read from memory.

INTA is NAND combined with DBIN (10) to provide system control signal $\overline{IBS}$, which is low or valid when the CPU has been interrupted and is expecting an interrupt vectoring single byte instruction (e.g. RST, Restart instruction) to be gated onto the data bus.

INP is NAND combined with DBIN (10) to provide system control signal $\overline{INP}$, which is low or valid when the CPU wants to input information from an input port.

OUT is NAND combined with the inverted $\overline{WR}$ signal (9) from the CPU to provide system control signal $\overline{OUT}$ which is low or valid when the CPU wants to output information to an output port.

The system control signal $\overline{OUT}$ is in turn NAND combined with WR to form system control signal $\overline{WRITE}$, which is valid or low when the CPU want to write into RAM memory.

The HOLDA signal from the CPU is double buffered (9) to provide HOLDA and $\overline{HOLDA}$ control signals. HOLDA or Hold Acknowledge is valid or high when the CPU has received a HOLD input high and has completed its current machine cycle. When the HOLDA signal goes high, all the address buffers for A0 thru A15 (6, 7, 8), will go into the high impedance output state. This isolates the CPU from the system address bus allowing DMA. Similarly when HOLDA is low, all the data bus buffers for D0 thru D7, (1, 2, 3) will go into the high impedance output state.

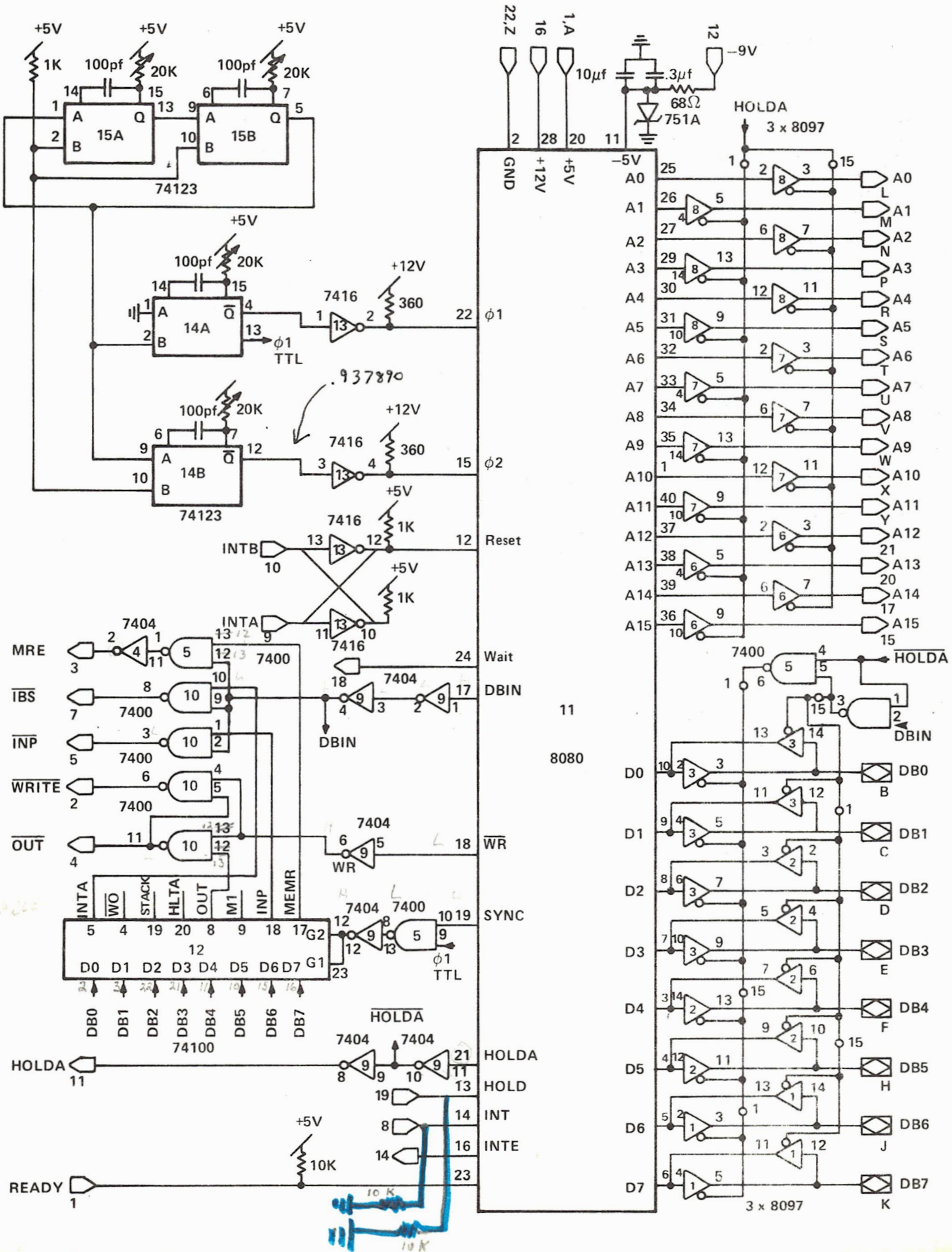
This isolates the CPU from the system data bus allowing DMA.

Whereas the address buffers (6, 7, 8) are 16Bit unidirectional out from the CPU, the data bus buffers (1, 2, 3) are 8Bit bidirectional. The direction of data flow through the data bus buffers is determined by the DBIN control signal. When DBIN is high, data flows from the system data bus into the CPU. When DBIN is low, data flows from the CPU onto the system data bus.

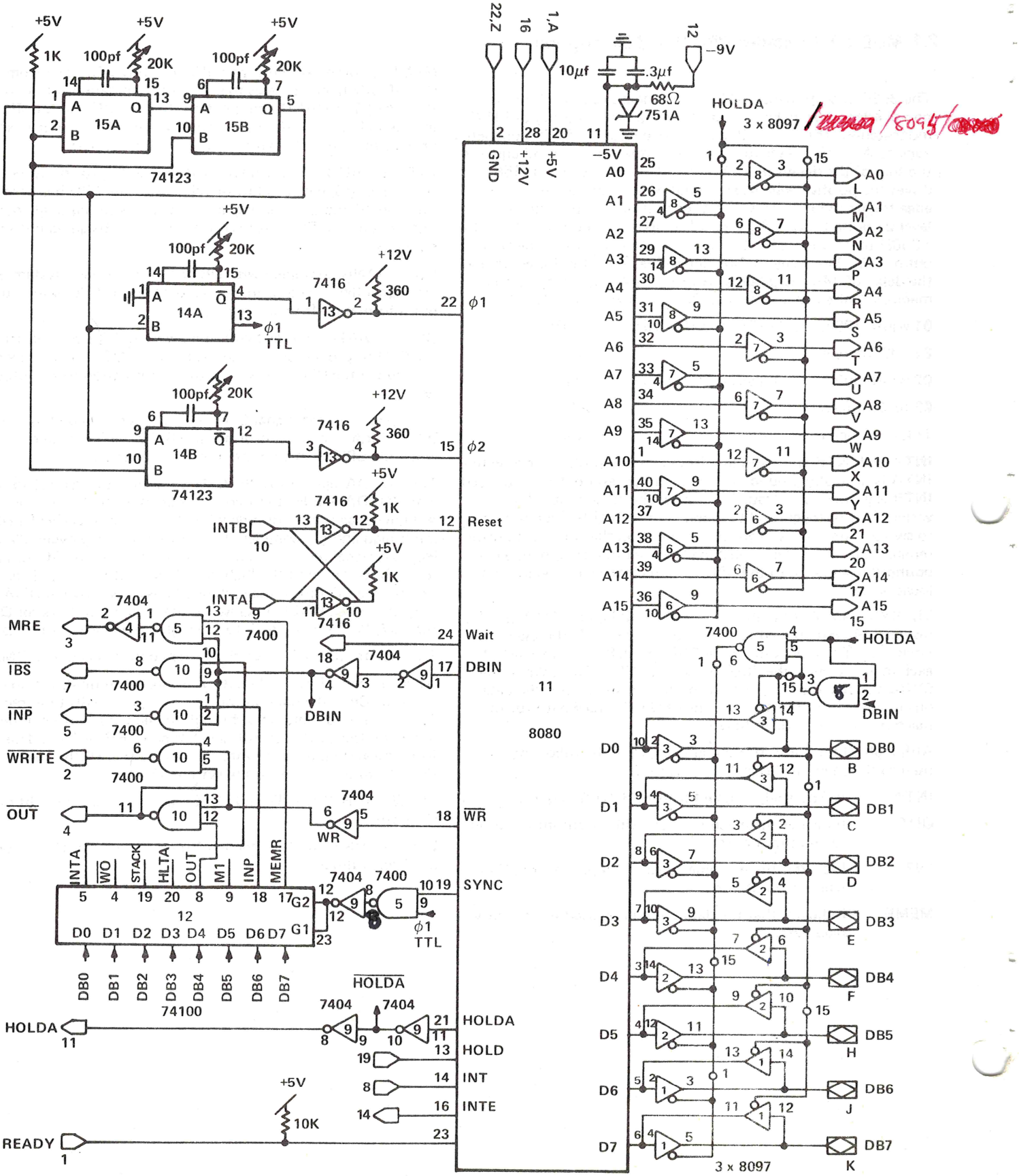
The WAIT, HOLD, INT and INTE signals are also brought off the CPU card.

The −5V VBB supply is derived from −9V via a 68Ω resistor and a 5V zener diode.

MOD 80-1 (CPU)
Circuit Schematic



MOD 80-1 (CPU)
Circuit Schematic



MOD80-1 8080 CPU Board

2	74123	Dual Monostable
1	74100	8 Bit Latch
1	7416	Quad Open Collector Inverter
1	8080	CPU
2	7400	Quad 2 Input Nand
2	7404	Hex Inverter
6	8097 (OR 74 367)	Hex Tri-state Buffer (SN 74367)
3		10μf electrolytic capacitor
1	IN751A	5-1v zener diode
1		68Ω ½ watt resistor
2		360Ω ½ watt resistor
1		10KΩ ¼ watt resistor
3		1KΩ ¼ watt resistor
4		20KΩ trimport
4		100pf capacitor
1		.33~~pf~~ uf capacitor

MOD 80-1 Printed Circuit Card

1 SEC. SCALE | .1 SCALE

FREQ

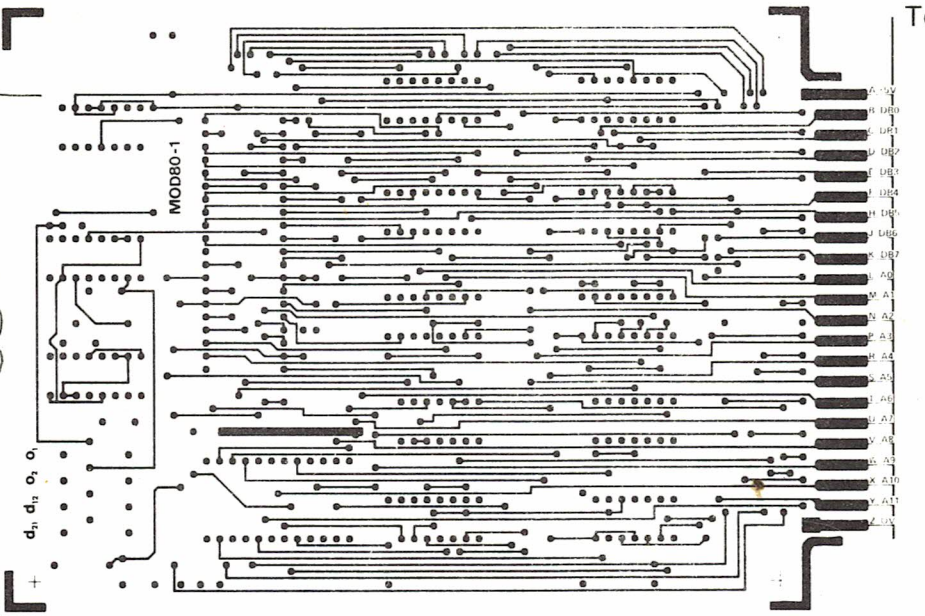
N1 = .936850 | .93761

~~N2 = .9377(56)~~

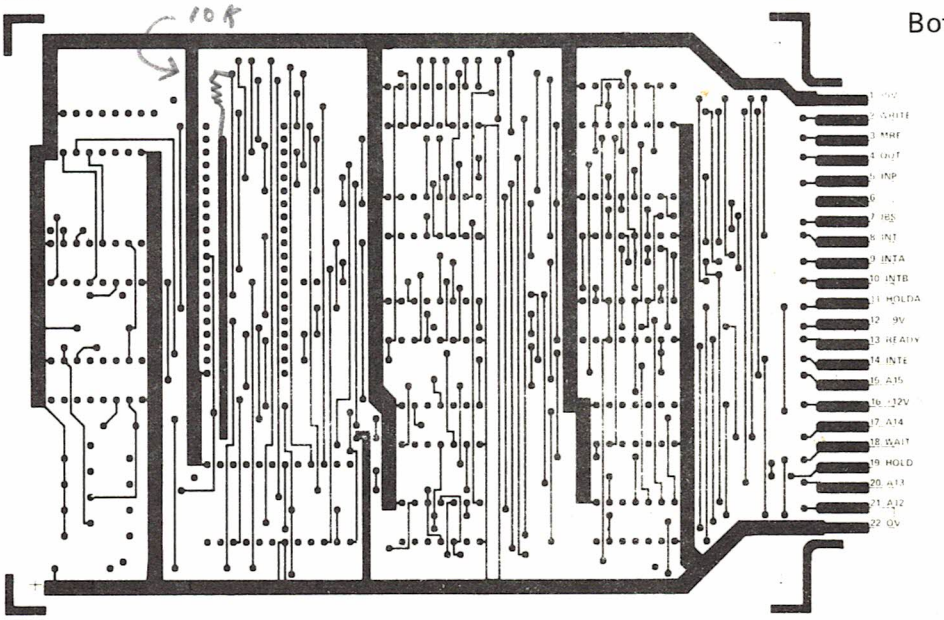
N2 = .9378(22) | .93786

N3 = ~~.1962XX~~ .196XXX (.1 SEC)

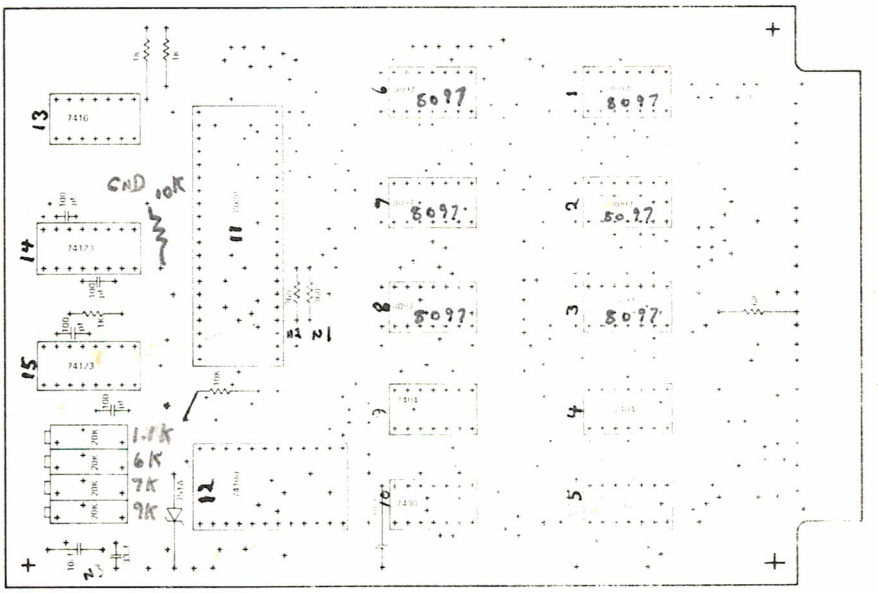
.206XXX (.1 SEC)



Top



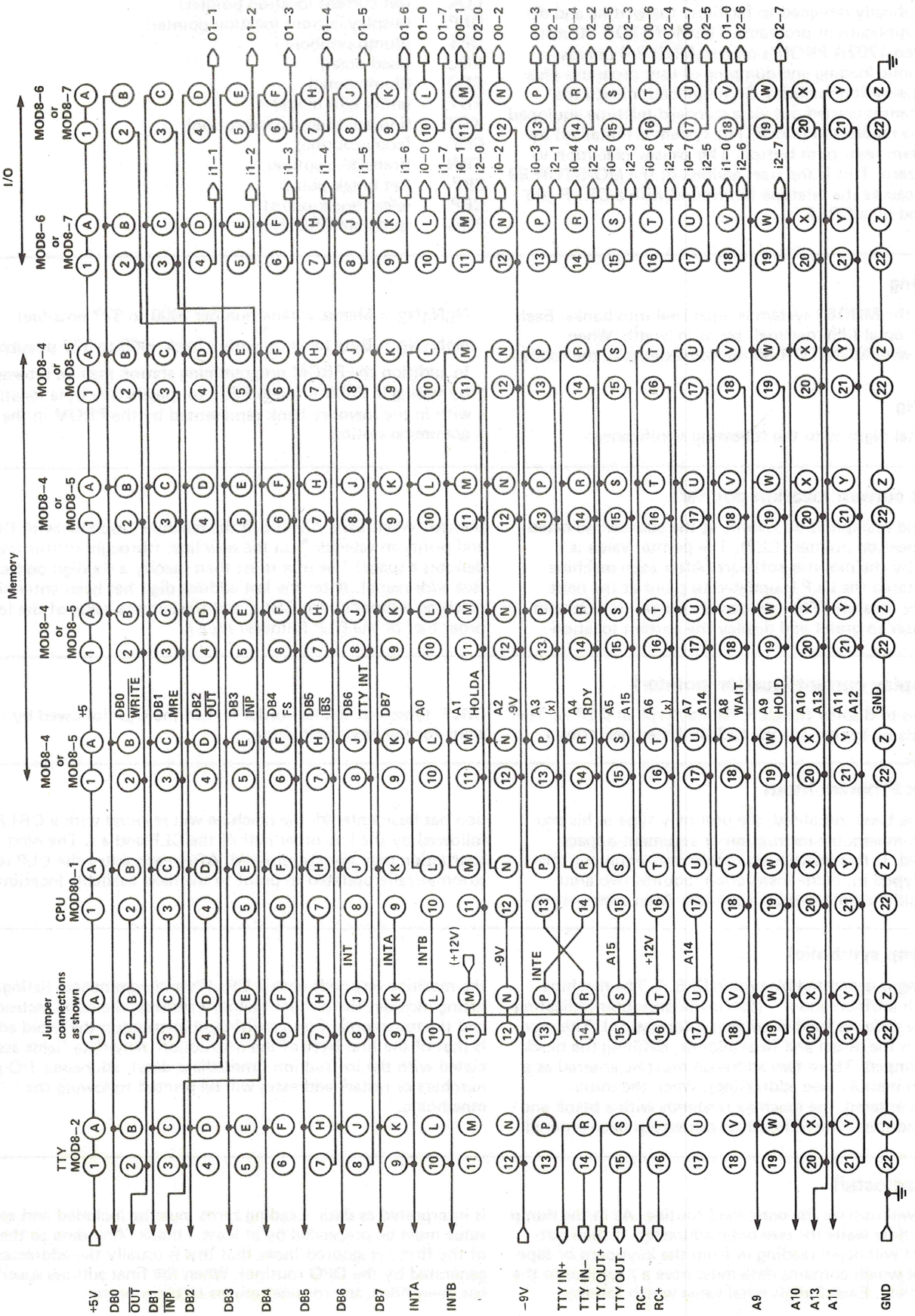
Bottom

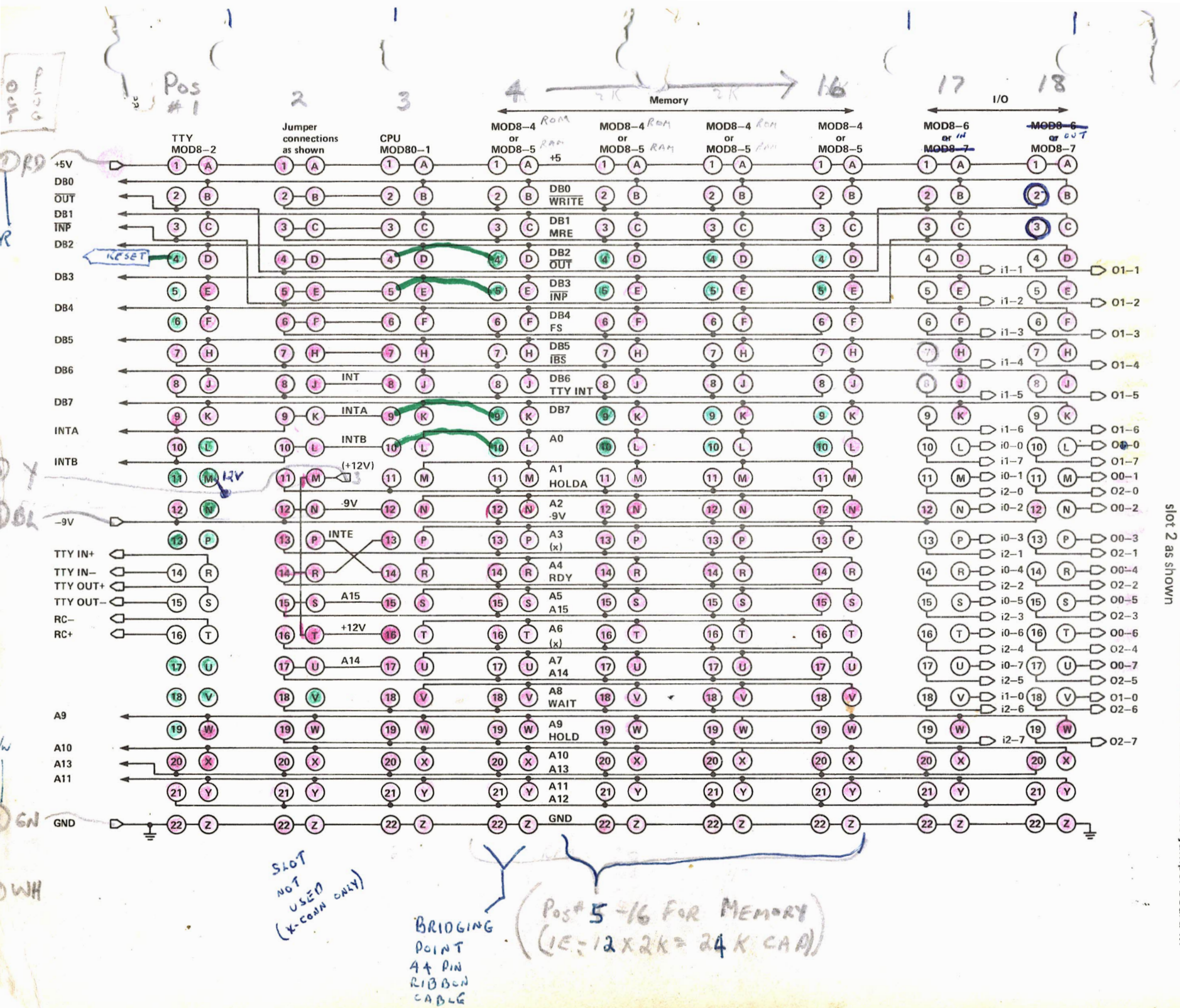


Components Layout

Backplane Configured for 8080 CPU

This is the same backplane as is used with the 8008 C.P.U., but the MOD80-1 CPU board resides in slot 3 and the jumper board in slot 2 as shown





TOP VIEW (OR FRONT VIEW)

(WHEN SEEN FROM REAR T.S. THE A-B-C IS ON LEFT AND 1-2-3-22 IS ON RIGHT SIDE 19 OR 22 PIN CONNECTOR)

Backplane Configured for 8080 CPU

This is the same backplane as is used with the 8008 C.P.U., but the MOD80-1 CPU board resides in slot 3 and the jumper board in slot 2 as shown

3. Monitor 80 Users Guide

The hardware configuration described above is not directly useable for system development without a compatible software package. The MONITOR 80 software package described in this section was specifically designed to facilitate the writing and debugging of user applications programs. The MONITOR 80 software resides in ten 1702A PROM's on two MOD 8-4 boards, and allows symbolic loading and dumping of user programs plus editing and manipulation facilities. Communication with the prototyping system is carried out via a standard teletype equipped with a paper tape reader and punch. The system is initialised by pressing the system reset push button. This causes execution to start at address zero. This is the start address of the MONITOR 80 software which causes the teletype to respond with a CRLF and 6 dashes followed by a CRLF.

The MONITOR 80 software is now ready to load symbolic program input or accept one of the following utility commands:

LOC	(set current location pointer)
DLP	(display current location pointer)
DPS	(dump symbolic)
LDO	(load octal)
DPO	(dump octal)
EDT	(enter edit mode)
XQT	(initiate program execution)
CPY	(copy routine)
TRN	(translate routine)
SBP	(set break-point)
CBP	(clear break-point)
PRG	(program PROM)

3.1 Addressing

The memory in the MOD80 system is organized into banks. Each bank is 0 to 377 octal (256 decimal) bytes in length. When communicating with MONITOR 80 the addresses take the following form:

$N_5N_4N_3N_2N_1N_0$

$N_0 - N_5$ are octal digits with the following significance:

$N_5N_4N_3$ = Memory bank number (000 to 377 possible)

$N_2N_1N_0$ = Byte location within bank (000 to 377 possible)

In addition the PROM programming station may be interogated by typing in P000 through P377 to specify any byte locations with in the memory bank represented by the PROM in the programming station.

3.2 LOC (set current location pointer)

All data entry and manipulation is done at the address indicated by the current location pointer (CLP). The pointer value is stored and used by the monitor software. After each machine instruction is entered the CLP is updated to point at the next available memory location. The two pseudo operators LOC and DLP allow the user to preset and display the current location pointer. When LOC Λ is typed the machine responds with CRLF and prints an asterisk * on the new line. (throughout this text Λ denotes a space) The user must then specify a six digit address (see addressing). After the last address digit has been entered machine responds with CRLF and the current value of the low order half of the CLP followed by a /.

3.3 DLP (display current location pointer)

If the user wishes to display the CLP, he may type in DLP Λ. The machine responds by typing out the CLP and then performs a CRLF types out the low order half of the CLP followed by / and waits for the next instruction.

3.4 Symbolic Program Input

Once the CLP has been initialised, the user may type in his program. After each mnemonic instruction or argument a space should be entered. If this instruction requires an argument this should now be typed in. If the instruction requires two arguments they should be separated by a comma. After each instruction has been entered, the machine will respond with a CRLF followed by the low order half of the CLP and a /. The next instruction may then be entered. After each entry the CLP is automatically updated to point to the next available location.

3.5 DPS (dump symbolic)

A symbolic listing is generated by typing DPS Λ. The machine will respond with a CRLF and a * (this is the prompter indicating that the machine requires further address information). The user must now type in the initial and final address, defining the block of code to be dumped. These two addresses must be entered as a 6 digit split octal number (see addressing). When the initial address has been entered, the machine responds with a blank and awaits the final address. When the final address has been entered the machine responds with 2 CRLF's and commences listing. The listing includes the current memory address, the octal instruction and the mnemonic. For a multi-byte instruction the listed address is that of the first byte of the instruction. Any data fields associated with the instruction (immediate data), addresses, I/O port numbers or restart addresses will be printed following the mnemonic.

3.6 LDO (load octal)

Typing LDO Λ will initiate the octal load routine. As in the dump routine, the machine waits for two octal addresses. It then outputs a CRLF and will begin reading in from the keyboard or tape reader. Each line which contains data must have a / symbol to the left of the data field. Each 3 digit octal value which follows/ is interpreted as data. Leading zeros must be included and each value must be preceeded by at least 1 blank. Any data to the left of the first / is ignored (note that this is usually the addresses generated by the DPO routine). When the final address specified has been filled, the routine returns to the monitor.

3.7 DPO (dump octal)

The dump routine will list 8 three digit octal values per line. Each line is started by the current address followed by a /. The user must specify the starting and ending address. When a DPO Λ is typed the machine will respond with a CRLF*. The first valid octal digit (0-7) typed will be interpreted as the beginning of the "initial address". (see addressing) After N_0 has been entered the machine will respond with a space. Next the ending address must be typed. After both addresses are entered the machine will start the dump. It will continue until it has typed the final location and then return to the controller.

3.8 EDT (enter edit mode)

The edit mode is entered by typing EDT. The editor responds with a CRLF*. The user must now enter the address at which he wishes to edit (CLP). The editor is now ready to accept one of the following commands:

nnn	where nnn is a three digit octal value to be loaded into memory
Λ	display memory value
↑	decrement the current location pointer
*AAAAAA	redefine the current location pointer with the value AAAAAA
@	equivalent to XQT
R	return to the monitor

If data is to be loaded it must immediately follow the / symbol. An invalid symbol will cause a CRLF with the CLP retyped. The nnn value is assembled as an 8 bit word and stored in the memory. Attempting to write a ROM address will not be flagged, yet the data will not (cannot) be written.

If a blank is entered after the / the current memory location will be displayed. Two options are then available:

a. nnn replace the current value with nnn

b. any other editor command

3.9 XQT (initiate program execution)

The XQT command allows the user to start the execution of his program. Following the typing of XQT Λ the machine will respond with CRLF* and wait for the starting address of the program. The user may return to monitor by including a RST 000 (restart) at the end of his routine.

3.10 CPY (copy routine)

Typing a CPY Λ will initiate a copy of a block of memory. The machine will respond with CRLF* and wait for the new start address where the block is to be copied to. The machine then responds with a further CRLF* and waits for the original start address and end address (defining the block to be moved). After the third address has been entered, the entire block specified will be copies unchanged starting at the new address. When the copy has been completed control returns to the monitor.
It should be noted that the copy may be made from one part of memory to another part of memory even though these parts overlap.

3.11 TRN (translate routine)

Typing TRN Λ when in the monitor mode initiates the translate. This routine is intended to allow relocation of a working program. After typing TRN Λ the machine will respond with CRLF* and wait for an address. This address indicates the beginning of the block of code to be operated on. After entry of this address the machine again responds with CRLF* and waits for an address. This address indicates the new start address to which the code will be relocated. The machine again responds with CRLF* and now waits for two further addresses. These addresses are the start and finish addresses from which the block of code originated.

3.12 SBP (set break-point)

Break-points allow the tracing of program flow during its execution. If a RST 010 command is encountered during program execution the monitor software will print out the contents of the A B C D E H and L registers, the memory contents addressed by the H and L registers and the CY2, P, CY1, Z , S flags.

The SBP command inserts a RST 010 command at the address specified by the user. The address at which the break-point is inserted and the instruction originally found there is retained by the monitor. Before setting subsequent break-points, the monitor will first restore data at the previous break-point location.

3.13 CBP (clear break-point)

The CBP command will restore the data at the present break-point location.

3.14 PRG (program PROM)

The monitor software also contains the facility for controlling a pROM programming station if one is attached to the system. The programming routine is entered by typing PRG Λ. The programming routine will allow programming a pROM with data presently located in memory. An initial and final address must be specified. The routine will program the data from specified location to the corresponding word location within the PROM .

e.g. 010177 Location 177 of the PROM

To accommodate 1602A/1702A devices it is necessary to change the programming duty cycle from 2% to 20%. After receiving the initial and final address to be programmed the programming routine will respond with CR/LF "%". The user must then type an "A" or "N" to determine the timing loop. Typing A will give a program pulse duty cycle of 20% for 1702A types and an "N" gives approximately 2% duty cycle for normal devices. There is no check for validity of the constants entered and under no condition should standard devices be programmed with excessive duty cycles.

The programming routine will first check if the PROM data is equal to the program data. If the byte patterns are identical the routine proceeds to the next address. If the location must be programmed the device is hit with a single program pulse and the data is again checked against the desired data. When the data is finally read as being valid, after B program pulses, the device is hit with an additional 4 x B program pulses.

It should be noted that attempting to program a standard device using 1602A/1702A timing (20% duty cycle) will destroy the device. Note also that the unprogrammed state for an A series device is all lows whereas for standard devices it is all highs.

Control A

Included in the TTY input routine is a check for the CTRLA k Depressing the CTRL button and A key simultaneously will cause the machine to immediately return to the monitor routine, and is equivalent to a monitor restart.

Rubout

Octal data input routines will accept a rubout command. Each time the rubout key is pressed a ← symbol is printed and a character is deleted. Typing two rubouts will delete two characters etc. The rubout routine for octal values will "back space" only to the beginning of the field. Data is presented by 1 field (or byte) whereas addresses are represented by two bytes (fields). The routine will type a ← for each rubout until it reaches the beginning of the field where it will accept a rubout but will not type any symbol and will not continue to back space.

The MONITOR 80 software with the facilities as outlined above uses locations 000000 through 007377 and 020000 through 021360 for ROM. In addition the MONITOR 80 software requires locations 010000 through 010077 as scratch pad RAM area.

The entire MONITOR 80 software package is listed below with comments.

---

4. Monitor 80 Software Structure and Listing

The MONITOR 80 software consists of three main parts. The first part from address 000000 to 003320 includes all the routines used by the octal editor.

The second part (005143 to 007362) consists of the symbolic routines i.e. the routines dealing with symbolic loading and dumping of the users program.

The last part consists of three independent routines: CPY (copy), TRN (translate) and PRG (Program PROM)

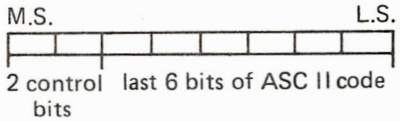
Pressing the reset button will initiate execution at address 000000. After clearing various flags and printing——————— MONITOR 80 waits for a character from the TTY. As characters are entered from the TTY, they are stored. The end of instruction or command is detected when a non-alphabetical character is entered.

As soon as this has happened, the system determines whether 2, 3 or 4 letters have been typed in, and tries to match these letters with the ASC II characters stored in the 2, 3 or 4 letter tables stored at locations 003321 to 005142.

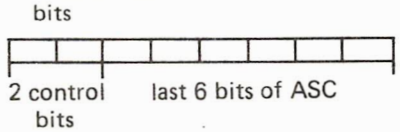
003327 – 004007 4 letter table
004010 – 004064 2 letter table
004065 – 004110 the argument table
004111 – 004216 the command table
004216 – 005142 3 letter table

The format of the 2, 3, 4 letter tables is as follows:–

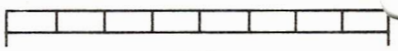
First letter



Second letter



Machine code



The two control bits hold information about the number of arguments in the following way:–

First letter: Ø – no argument
1 – 1 argument, destination
2 – 2 argument
3 – 1 argument, source

Second letter: Ø – no data byte
1 – 1 data byte
2 – 2 data bytes

Third letter: Ø – normal instruction
1 – exception
2 – 4 letter instruction
3 – 4 letter instruction and exception

Exceptions are DAD, DCX, and LDAX instructions.

Command table format is as follows for these:–

First letter	ASC II
Second letter	ASC II
Third letter	ASC II
	L.S. address byte
	M.S. address byte

These determine the address where the command routine is stored.

If the system does not find a match between the instruction or command typed in, and the ASC II characters within the table, it will print "? " and wait for a new entry.

4.1 Interrupts

Six vectored interrupts are available to the user. They are activated by jamming a RST 020 thru RST 070 instruction on the Data Bus during an acknowledged interrupt.

Suppose a RST 020 instruction is jammed on the Data Bus. The Program Counter is stored in the Stack, the Stack is pushed down by two bytes, and the Program Counter is set to 000020. So, the program was interrupted and now continues execution at address 000020. There it sees the call to the Save routine, so all registers are saved in the Stack. Then the H,L registers are loaded with the contents of locations 010000 and 010001, which should contain the address of the Interrupt service routine. The next instruction is an indirect jump to the location pointed at by H,L. The last instruction of the Interrupt service routine should be Call 000373 i.e. call the Restore routine. This will load all registers and flags with their original values. A RET instruction will then return control to the Interrupted program.

Locations where the addresses of the Interrupt service routines are stored:—

RST 020	010100, 010101
RST 030	010102, 010103
RST 040	010104, 010105
RST 050	010106, 010107
RST 060	010110, 010111
RST 070	010112, 010113

This implies that the memory available to the user starts at 010114 if the Interrupts are used, and 010100 if no Interrupts are used. Locations 010000 to 010077 are used by MONITOR 80.

These locations are used as follows:

Address		Routine which uses it
010000 010001	CLP, L.L	Many Routines
010002 010003	U.L	Many Routines
010004 010005	New Range	CPY, TRN
010006 010007	Where the program is now	TRN only
010007	Also PROM programmer buffer	Set H,L to CLP routine
010010	I/O flag; TTY is 000	Input TTY routine
010011	Exception flag	DPS, LDS, TRN
010012 010013	MMM+(U.L-L.L)	TRN Routine
010013	Also I/O Buffer	TTY O/P
010014 010015	A,L store	Save Routine
010014	Also move up flag	TRN Routine
010016 010016	PROM Programmer flag Also code in TRN	Many Routines
010017 010020	Breakpoint address	SBP, CBP routine
010021	Breakpoint — byte stored	
010022	Breakpoint flag	
010023	Data I/P buffer	LDS routine
010024	RST byte buffer	LDS routine
010023 010024	U.L-L.L	TRN routine
010027	Argument flag	LDS routine
010030	Table Flag	LDS routine
010031 010034	Empty	Possible S.P.
010035)))) 010077)	S.P.	

4.2 Subroutine Index

List of useful subroutines used by MONITOR 80 which are available to the user.

Start Address	Name	Description	Remark
000357	Save	Will save the contents of all registers plus flags in the stack	Uses the location 010014 010015
000373	Restore	Will restore contents of all registers and flags	
001001	TTY In	Will input the character from TTY into A register	Uses 010013
001117	Timing Loop	General purpose delay	
001133	TTY Out	Output the character in A register on TTY	Uses 110013 Check 010010
001241	I/P Byte	Inputs octal byte to memory addressed by H, L	
001313	O/P Space	Outputs blank character on TTY	
001323	O/P Bytes	Outputs the octal contents of memory addressed by H, L onto TTY	
001355	O/P CR/LF	Output carriage Return/Line Feed	
001372	O/P /	Output /	
003260	O/P with Loop	A registers hold ASCII code, B registers hold number of repetitions	

The following is a complete commented listing of the MONITOR 80 software package.

```
000000/ 076 MVI   A   001          IDLE TTY
000002/ 323 ØUT       024          -
000004/ 303 JMP       000100       CØNTINUE ELSWHERE
000007/ 000 NØP
000010/ 345 PUSH  H                SAVE H,L.      RST 010;BRK.PT.EXEC
000011/ 041 LXI   H   010023       SET TØ PRINT BUFFER(PB)
000014/ 167 MØV   M A              ACC TØ PB
000015/ 303 JMP       003160       CØNT ELSWHERE
000020/ 315 CALL      000357       SAVE.          RST 020
000023/ 052 LHLD      010100       INDIRECT JUMP TØ LØC PØINTED
000026/ 351 PCHL                   BY CØNTENT ØF 010100,010101
000027/ 000 NØP
000030/ 315 CALL      000357                      RST 030
000033/ 052 LHLD      010102       IND.JMP.PØINTED BY
000036/ 351 PCHL                   010102,010103
000037/ 000 NØP
000040/ 315 CALL      000357                      RST 040
000043/ 052 LHLD      010104
000046/ 351 PCHL
000047/ 000 NØP
000050/ 315 CALL      000357                      RST 050
000053/ 052 LHLD      010106
000056/ 351 PCHL
000057/ 000 NØP
000060/ 315 CALL      000357                      RST 060
000063/ 052 LHLD      010110
000066/ 351 PCHL
000067/ 000 NØP
000070/ 315 CALL      000357                      RST 070
000073/ 052 LHLD      010112
000076/ 351 PCHL
000077/ 000 NØP
000100/ 323 ØUT       026          DISABLE R.C.,STØP PRØGRAMMER
000102/ 041 LXI   H   010100       SET S.P
000105/ 371 SPHL                   -
000106/ 076 MVI   A   000          CLEAR TTY I/Ø,PRG,
000110/ 062 STA       010010       EXCEPTIØN FLAGS
000113/ 062 STA       010011
000116/ 062 STA       010016       -
000121/ 076 MVI   A   055          Ø/P ------
000123/ 006 MVI   B   006
```

(OUTPUT WITH LOOP)

```
000125/ 315 CALL      003260      -
000130/ 315 CALL      001355      CR/LF
000133/ 315 CALL      003277      I/P NØNBLANK,FIRST LETTER
000136/ 312 JZ        000133      -
000141/ 346 NDI       077         MASK
000143/ 117 MØV   C A             STØRE IN C
000144/ 315 CALL      003277      I/P ALPHABETIC,SECØND LETTER
000147/ 322 JNC       006070      -
000152/ 346 NDI       077         MASK
000154/ 127 MØV   D A             STØRE IN D
000155/ 315 CALL      003277      I/P ALPHABETIC,THIRD LETTER
000160/ 322 JNC       000250      -
000163/ 346 NDI       077         MASK
000165/ 137 MØV   E A             STØRE IN E
000166/ 315 CALL      003277      I/P ALPHABETIC,FØURTH LETTER-
000171/ 322 JNC       000221      -
000174/ 315 CALL      003277      I/P NØNALPHABETIC
000177/ 332 JC        006070      -
000202/ 041 LXI   H   003321      SET H,L TØ 4 LET.TABLE.    4 LET.
000205/ 006 MVI   B   013         INSTR CØUNT                DEC.
000207/ 315 CALL      005166      SWEEP 4 LETTER TABLE     CØNTRØLER
000212/ 043 INX   H               NØ MATCH,GØ TØ NEXT
000213/ 302 JNZ       000207      SET ØF BYTES
000216/ 303 JMP       006070      END ØF 4 LET.TABLE,ERRØR
000221/ 041 LXI   H   004111      SET H,L TØ BEG.ØF CØMMAND TABLE
000224/ 006 MVI   B   016         SET UP CØUNT
000226/ 315 CALL      000270      CALL 3 LET.MATCH,IF NØ
000231/ 043 INX   H               MATCH,GØ TØ NEXT BYTE
000232/ 302 JNZ       000226      NØT END ØF TABLE,TRY AGAIN
000235/ 006 MVI   B   065         NØT CØMMAND,SET CØUNT TØ 3 LET INS.T
000237/ 315 CALL      000300      CALL 3 LET INST MATCH
000242/ 302 JNZ       000237      NØT END,TRY AGAIN
000245/ 303 JMP       006070      END ØF TABLE,ERRØR
000250/ 041 LXI   H   004010      SET H,L TØ 2 LET.TABLE
000253/ 006 MVI   B   017         SET UP CØUNT
000255/ 132 MØV   E D             SHIFT
000256/ 121 MØV   D C             -
000257/ 315 CALL      000306      CALL 2 LET.MATCH
000262/ 302 JNZ       000257      NØ MATCH,TRY AGAIN
000265/ 303 JMP       006070      END ØF TABLE,ERRØR
000270/ 315 CALL      000314      CALL 3 LET.MATCH          EXECUTE
000273/ 136 MØV   E M             MATCH FØUND,LØAD          GIVEN
000274/ 043 INX   H               JUMP ADDR.TØ D,E          CØMMAND
000275/ 126 MØV   D M             -
000276/ 353 XCHG                  INDIRECT JMP
000277/ 351 PCHL                  -
000300/ 315 CALL      000314      CALL 3 LET.MATCH
000303/ 303 JMP       005143      MATCH FØUND,CØNT ELSWHERE
000306/ 315 CALL      000323      2 LET MATCH
000311/ 303 JMP       005175      MATCH FØUND,CØNT ELSWHERE
000314/ 315 CALL      000352      MASK,INCREMENT H,L
000317/ 271 CMP   C               THIRD LETTER EQUAL TØ TAB?
000320/ 302 JNZ       000343      NØ,JMP
000323/ 315 CALL      000352      YES,GØ TØ SECØUND LETTER
000326/ 272 CMP   D               SECØUND LETTER EQUAL?
000327/ 302 JNZ       000344      NØ
000332/ 315 CALL      000352      YES,GØ TØ FIRST LETTER
000335/ 273 CMP   E               LAST LETTER EQUAL?
000336/ 302 JNZ       000345      NØ
000341/ 136 MØV   E M             XES,MØVE CØDE TØ E
000342/ 311 RET
000343/ 043 INX   H
000344/ 043 INX   H
000345/ 043 INX   H
000346/ 063 INX   SP              SET SP
000347/ 063 INX   SP              -
000350/ 005 DCR   B               CØUNT INSTR.
000351/ 311 RET
000352/ 076 MVI   A   077                   MASK
```

```
                                                    SUBRØUTINE
000354/ 246 NDA   M
000355/ 043 INX   H
000356/ 311 RET                       -
000357/ 042 SHLD      010014          SAVE H,L TEMPØRARILY          SAVE
000362/ 343 XTHL                      EXCH.H,L AND SP               SUBR.
000363/ 365 PUSH  PSW                 SAVE REGISTERS
000364/ 305 PUSH  B
000365/ 325 PUSH  D
000366/ 345 PUSH  H
000367/ 052 LHLD      010014          RESTØRE H,L
000372/ 311 RET
000373/ 341 PØP   H                                         RESTØRE
000374/ 321 PØP   D                                         SUBR.
000375/ 301 PØP   B
000376/ 361 PØP   PSW
000377/ 343 XTHL
001000/ 311 RET                       -
001001/ 315 CALL      000357          SAVE REG              I/P TTY
001004/ 072 LDA       010010          TTYI/P?               SUBR.
001007/ 376 CPI       000
001011/ 302 JNZ       000000          -
001014/ 015 MVI   C   367             BIT CØUNT
001016/ 074 INR   A                   ENABLE R.C.
001017/ 323 ØUT       026             -
001021/ 333 IN        000             WAIT FØR START PULSE
001023/ 017 RRC
001024/ 322 JNC       001021          -
001027/ 076 MVI   A   000             DISABLE R.C.
001031/ 323 ØUT       026             -
001033/ 041 LXI   H   001040          SET TIME CØNSTANT(T.C.)FOR 110 BAUD.
001036/ 315 CALL      001117          TIMEØUT 4.05 MSEC
001041/ 051 DAD   H                   DØUBLE T.C.
001042/ 333 IN        000             I/P BIT
001044/ 057 CMA                       CØMPLEMENT
001045/ 323 ØUT       024             ECHØ
001047/ 037 RAR                       RØT.INTØ PØSITIØN
001050/ 170 MØV   A B
001051/ 037 RAR
001052/ 107 MØV   B A                 -
001053/ 315 CALL      001117          TIMEØUT 9.1 MSEC
001056/ 014 INR   C                   CØUNT BIT
001057/ 302 JNZ       001042          NØT LAST,LØØP
001062/ 076 MVI   A   001             IDLE TTY
001064/ 323 ØUT       024             -
001066/ 051 DAD   H                   DØUBLE T.C.
001067/ 315 CALL      001117          TIMEØUT 18.2 MSEC
001072/ 170 MØV   A B                 MASK PARITY
001073/ 346 NDI       177             -
001075/ 062 STA       010013          PUT INTØ TEMP.STØRE
001100/ 315 CALL      000373          RESTØRE REG.
001103/ 072 LDA       010013          TEST FØR TAPE LEADER
001106/ 376 CPI       000             -
001110/ 312 JZ        001001
001113/ 376 CPI       001             TEST FØR CTRL A
001115/ 300 RNZ
001116/ 307 RST       000             IF CTRL A,GØ TØ BEG.ØF MØN.
001117/ 345 PUSH  H                                         TIMING
001120/ 044 INR   H                                         LØØP
001121/ 055 DCR   L
001122/ 302 JNZ       001121
001125/ 045 DCR   H
001126/ 302 JNZ       001121
001131/ 341 PØP   H
001132/ 311 RET
001133/ 315 CALL      000357          SAVE REG.             TTY Ø/P
001136/ 016 MVI   C   370             CØUNTER
001140/ 062 STA       010013          MØVE TØ I/Ø BUFFER
001143/ 072 LDA       010010          TTY I/P TEST
001146/ 376 CPI       000
001150/ 302 JNZ       000000
001153/ 323 ØUT       024             START TTY
```

CALL 001327 (HI)
021327 (LOW)

USE 000152 for 300 BAUD

000063 for 600 BAUD

CALL 201327

000063 for 600 BAUD
000152 for 300 BAUD
001040 for 110 BAUD.

```
001155/ 041 LXI   H  001040     SET T.C. (for 110 BAUD) USE
001160/ 051 DAD   H             DØUBLE T.C.
001161/ 072 LDA      010013     RESTØRE ACC
001164/ 315 CALL     001117     TIMEØUT 9.1 MSEC
001167/ 323 ØUT      024        Ø/P BIT
001171/ 037 RAR                 RØTATE
001172/ 014 INR   C             CØUNT
001173/ 302 JNZ      001164     CØNTINUE ØUTPUTING
001176/ 315 CALL     001117     TIMEØUT 9.1 MSEC
001201/ 076 MVI   A  001        IDLE TTY
001203/ 323 ØUT      024        -
001205/ 051 DAD   H             TIMEØUT 18.2 MSEC
001206/ 315 CALL     001117     -
001211/ 315 CALL     000373     RESTØRE REG
001214/ 311 RET
001215/ 315 CALL     001001                          I/P WITH
001220/ 107 MØV   B A                                ØCTAL TEST
001221/ 346 NDI      370                             INTØ B
001223/ 376 CPI      060
001225/ 311 RET                 -
001226/ 315 CALL     001001                          I/P WITH
001231/ 376 CPI      177                             RUBØUT
001233/ 300 RNZ
001234/ 076 MVI   A  137        PRINT
001236/ 303 JMP      001133     -
001241/ 315 CALL     000357     SAVE REG.            I/P BYTE
001244/ 315 CALL     001215     I/P ØCT              TØ MEMØRY
001247/ 302 JNZ      001244     -
001252/ 170 MØV   A B           RØTATE,SAVE IN B
001253/ 017 RRC
001254/ 017 RRC
001255/ 346 NDI      300
001257/ 107 MØV   B A           -
001260/ 315 CALL     001226     I/P ASCII WITH RUBØUT TEST
001263/ 312 JZ       001244     -
001266/ 346 NDI      007        MASK,RØTATE,SAVE IN C
001270/ 007 RLC
001271/ 007 RLC
001272/ 007 RLC
001273/ 117 MØV   C A           -
001274/ 315 CALL     001226     I/P ASCII WITH RUBØUT TDST
001277/ 312 JZ       001260     -
001302/ 346 NDI      007        MASK CØMBINE,
001304/ 260 ØRA   B             MØVE TØ MEMØRY
001305/ 261 ØRA   C
001306/ 167 MØV   M A           -
001307/ 315 CALL     000373     RESTØRE REG.
001312/ 311 RET
001313/ 365 PUSH  PSW                               Ø/P BLANC
001314/ 076 MVI   A  040
001316/ 315 CALL     001133
001321/ 361 PØP   PSW
001322/ 311 RET                 -
001323/ 315 CALL     001313     Ø/P BL               Ø/P MEMØRY
001326/ 176 MØV   A M           Ø/P FIRST DIGIT      BYTE
001327/ 007 RLC
001330/ 007 RLC
001331/ 346 NDI      003
001333/ 315 CALL     001350     -
001336/ 176 MØV   A M           Ø/P SEC.DIGIT
001337/ 017 RRC
001340/ 017 RRC
001341/ 017 RRC
001342/ 315 CALL     001346     -
001345/ 176 MØV   A M           Ø/P LAST DIGIT
001346/ 346 NDI      007
001350/ 366 ØRI      060
001352/ 303 JMP      001133     -
001355/ 365 PUSH  PSW                               Ø/P CR/LF
001356/ 076 MVI   A  015
001360/ 315 CALL     001133
001363/ 076 MVI   A  012
```

JUMP 022042

```
001365/ 315 CALL      001133
001370/ 361 PØP   PSW
001371/ 311 RET                    -
001372/ 365 PUSH  PSW                                  Ø/P /(SLASH)
001373/ 076 MVI   A   057
001375/ 315 CALL      001133
002000/ 361 PØP   PSW
002001/ 311 RET                    -
002002/ 052 LHLD      010000       SET TØ CLP.        SET H,L TØ CLP
002005/ 072 LDA       010016       TEST FØR PRG.
002010/ 376 CPI       001
002012/ 300 RNZ                    -
002013/ 175 MØV   A L              YES,PRG.SET TØ PRG.BUFFER
002014/ 041 LXI   H   010007       -
002017/ 323 ØUT       020          Ø/P ADDRESS
002021/ 333 IN        002          I/P BYTE
002023/ 167 MØV   M A              MØVE TØ BUFFER
002024/ 311 RET
002025/ 315 CALL      002002       SET H,L TØ CLP.     Ø/P CLP
002030/ 041 LXI   H   010001       SET TØ M.S.BYTE ØF CLP
002033/ 302 JNZ       002046       NØT PRG,JUMP
002036/ 076 MVI   A   120          Ø/P P(PRØGRAMMER)
002040/ 315 CALL      001133       -
002043/ 303 JMP       002051
002046/ 315 CALL      001323       PRINT M.S.BYTE ØF ADDRESS
002051/ 053 DCX   H
002052/ 303 JMP       001326       PRINT L.S.BYTE ØF ADDRESS
002055/ 315 CALL      001355                           Ø/P CLP/
002060/ 315 CALL      002025
002063/ 303 JMP       001372       -
002066/ 315 CALL      002002                           Ø/P MEMØRY AT CLP
002071/ 303 JMP       001323       -
002074/ 052 LHLD      010000                           INCREMENT CLP
002077/ 043 INX   H
002100/ 042 SHLD      010000
002103/ 303 JMP       002002       -
002106/ 052 LHLD      010000                           DECREMENT CLP
002111/ 053 DCX   H
002112/ 042 SHLD      010000
002115/ 303 JMP       002002       -
002120/ 052 LHLD      010000                           CØMPARE CLP
002123/ 353 XCHG                                       AND UPPER LIMIT
002124/ 052 LHLD      010002
002127/ 172 MØV   A D
002130/ 274 CMP   H
002131/ 300 RNZ
002132/ 173 MØV   A E
002133/ 275 CMP   L
002134/ 311 RET                    -
002135/ 315 CALL      002120       CØMPARE CLP AND U.L.     RANGE
002140/ 302 JNZ       002074       NØT ZERØ,INCR.CLP       CØNTRØL
002143/ 307 RST       000          END ØF PRØGRAMM,GØ TØ BEG.
002144/ 315 CALL      001355                           Ø/P CLP*
002147/ 076 MVI   A   052
002151/ 315 CALL      001133       -
002154/ 315 CALL      001313       Ø/P BLANC          I/P ADDR.TØ MEM
002157/ 315 CALL      001001       I/P ASCII
002162/ 376 CPI       120          =P?
002164/ 312 JZ        002210       YES SET PRG FLAG
002167/ 001 LXI   B   002204       NØ,SET UP NEXT ADDR.-I/P M.S.BYTE
002172/ 305 PUSH  B
002173/ 315 CALL      000357       SAVE REG.
002176/ 315 CALL      001220       I/P ØCTAL
002201/ 303 JMP       001247       -
002204/ 053 DCX   H                I/P BYTE TØ MEM.-I/P L.S.BYTE
002205/ 303 JMP       001241       -
002210/ 076 MVI   A   001          SET PRØGRAMMER FLAG
002212/ 062 STA       010016       -
002215/ 167 MØV   M A              STØRE P
002216/ 303 JMP       002204       I/P WØRD
002221/ 041 LXI   H   010007       I/P LØC WHERE PRØG.      I/P LIMITS
```

```
002224/ 315 CALL       002144      IS NØW
002227/ 041 LXI    H   010005      I/P NEW LØC.
002232/ 315 CALL       002144      -
002235/ 315 CALL       002251      Ø/P CR/LF*,I/P NEW CLP
002240/ 041 LXI    H   010003      Ø/P BL.,I/P UPPER L.
002243/ 315 CALL       002154      -
002246/ 303 JMP        001355      Ø/P CR/LF,RET
002251/ 041 LXI    H   010001      SET H,L TØ CLP          DPØ
002254/ 303 JMP        002144      Ø/P CR/LF*,I/P CLP
002257/ 315 CALL       002235      I/P UPPER L.,Ø/P CR/LF
002262/ 016 MVI    C   010         SET UP CØUNT
002264/ 315 CALL       002055      Ø/P CLP/
002267/ 315 CALL       002002      SET H,L TØ CLP
002272/ 315 CALL       001323      Ø/P MEMØRY BYTE
002275/ 315 CALL       002135      INCR CLP
002300/ 015 DCR    C               CØUNT
002301/ 302 JNZ        002272      LØØP
002304/ 303 JMP        002262      NEW LINE
002307/ 315 CALL       002235      I/P CLP,UPPER L.,Ø/P CR/LF      LDØ
002312/ 315 CALL       001001      I/P ASCII
002315/ 376 CPI        057         WAIT FØR /
002317/ 302 JNZ        002312      -
002322/ 315 CALL       002002      SET H,L TØ CLP
002325/ 315 CALL       001001      I/P ASCII,TEST
002330/ 376 CPI        015         FØR CR/LF
002332/ 312 JZ         002312      -
002335/ 315 CALL       001241      I/P BYTE TØ MEM.
002340/ 315 CALL       002135      RANGE CØNTRØL (END?)
002343/ 303 JMP        002325      LØØP
002346/ 315 CALL       002251      I/P CLP                 ØCTAL
002351/ 315 CALL       002055      Ø/P CLP                 EDITØR
002354/ 315 CALL       002002      SET H,L TØ CLP
002357/ 315 CALL       002365      PRØCESS LINE
002362/ 303 JMP        002351      LØØP
002365/ 315 CALL       001001      I/P ASCII               EDITING
002370/ 376 CPI        122         R?                      RØUTINES
002372/ 312 JZ         000000      YES,GØ TØ BEG.
002375/ 376 CPI        052         *?
002377/ 312 JZ         002251      YES,GØ TØ I/P CLP RØUTINE
003002/ 376 CPI        100          @?
003004/ 312 JZ         003073      XES,GØ TØ XQT
003007/ 376 CPI        136         ↑?
003011/ 312 JZ         002106      YES,GØ TØ DECR.CLP
003014/ 376 CPI        040         BLANC?
003016/ 312 JZ         003042      YES,PRINT ØCTAL BYTE
003021/ 315 CALL       001220      TEST FØR ØCTAL
003024/ 300 RNZ                    NØT ØCTAL
003025/ 021 LXI    D   003037      I/P BYTE TØ MEM.
003030/ 325 PUSH   D
003031/ 315 CALL       000357
003034/ 303 JMP        001252
003037/ 303 JMP        003045
003042/ 315 CALL       002066      Ø/P MEM AT CLP
003045/ 315 CALL       001001      I/P ASCII
003050/ 376 CPI        137         BACK ARRØW?
003052/ 302 JNZ        003063      NØ,JUMP
003055/ 315 CALL       001241      YES,I/P ANØTHER BYTE
003060/ 303 JMP        003045      LØØP
003063/ 376 CPI        040         BLANC?
003065/ 312 JZ         002074      YES,INCR.CLP AND RET.
003070/ 303 JMP        002370      NØ.LØØP
003073/ 315 CALL       001355      CR/LF                   XQT
003076/ 315 CALL       002251      I/P ADDRES TØ H,L
003101/ 052 LHLD       010000      -
003104/ 351 PCHL                   INDIRECT JUMP
003105/ 315 CALL       003135      CLEAR BPT           SET BRKPØINT
003110/ 041 LXI    H   010020      SET H,L TØ BPT ADDRESS BUFFER
003113/ 315 CALL       002144      I/P BPT ADDR.TØ MEM.
003116/ 052 LHLD       010017      SAVE PRØGR.BYTE
003121/ 176 MØV    A M
003122/ 062 STA        010021      -
```

```
003125/ 066 MVI    M    317          INSERT RST 010 INTØ PRØG.
003127/ 076 MVI    A    001          SET BPT FLAG
003131/ 062 STA         010022
003134/ 307 RST         000          -
003135/ 072 LDA         010022       TEST FØR BPT          CLEAR BPT
003140/ 376 CPI         001
003142/ 300 RNZ                      -
003143/ 052 LHLD        010017       LØAD H,LWITH BPT ADDR.
003146/ 072 LDA         010021       MØV PRØG BYTE TØ ACC
003151/ 167 MØV    M A               MØVE BACK INTØ PRØGRAMM
003152/ 076 MVI    A    000          CLEAR BPT  FLAG
003154/ 062 STA         010022
003157/ 311 RET                      -
003160/ 365 PUSH   PSW               SAVE ACC,FLAGS.       EXEC.BPT
003161/ 315 CALL        001323       Ø/P ACC
003164/ 160 MØV    M B               Ø/P B
003165/ 315 CALL        001323       -
003170/ 161 MØV    M C               Ø/P C
003171/ 315 CALL        001323       -
003174/ 162 MØV    M D               Ø/P D
003175/ 315 CALL        001323       -
003200/ 163 MØV    M E               Ø/P E
003201/ 315 CALL        001323       -
003204/ 301 PØP    B                 RESTØRE FLAGS,ACC TØ B,C
003205/ 321 PØP    D                 RESTØRE H,L TØ D,E
003206/ 162 MØV    M D               Ø/P H
003207/ 315 CALL        001323       -
003212/ 163 MØV    M E               Ø/P L
003213/ 315 CALL        001323       -
003216/ 353 XCHG                     Ø/P MEM.
003217/ 315 CALL        001323       -
003222/ 171 MØV    A C               FLAGS TØ ACC
003223/ 315 CALL        003250       Ø/P CARRY
003226/ 315 CALL        003244       Ø/P PARITY
003231/ 315 CALL        003244       Ø/P CY1(CARRY BETWEEN D3,D4)
003234/ 315 CALL        003244       Ø/P ZERØ
003237/ 171 MØV    A C               Ø/P SIGN
003240/ 315 CALL        003246
003243/ 307 RST         000          -
003244/ 171 MØV    A C                                SHIFT ARØUND
003245/ 017 RRC                                       FLAG BYTE
003246/ 017 RRC
003247/ 117 MØV    C A
003250/ 346 NDI         001
003252/ 315 CALL        001313
003255/ 303 JMP         001346       -
003260/ 315 CALL        001355       CR/LF                 Ø/P WITH
003263/ 315 CALL        001133       Ø/P ASCII             CØUNT LØØP
003266/ 005 DCR    B                 CØUNT
003267/ 310 RZ
003270/ 303 JMP         003263       LØØP
003273/ 315 CALL        003135       CLEAR BPT
003276/ 307 RST         000          -
003277/ 315 CALL        001001       I/P ASCII             I/P NUMERIC
003302/ 376 CPI         040          BLANC?                ALPHABETIC,BLANC
003304/ 305 PUSH   B                 SAVE B,C              TEST
003305/ 107 MØV    B A
003306/ 007 RLC                      RØTATE M.S.BIT INTØ CARRY
003307/ 007 RLC
003310/ 170 MØV    A B
003311/ 301 PØP    B                 RESTØRE B,C
003312/ 311 RET                      -
003313/ 076 MVI    A    077                           ERRØR,Ø/P ?
003315/ 315 CALL        001133                         AND RESTART
003320/ 307 RST         000          ----------
     FØLLØWING ARE VARIØUS LØØK UP TABLES
     (SEE MØNITØR STRUCTURE)

003321/ 003 201 214 014 315 120 025 223
003331/ 010 375 030 003 210 007 353 023
003341/ 210 214 004 042 014 210 214 004
```

```
003351/ 052 123 024 201 030 072 114 004
003361/ 301 030 072 030 003 210 007 353
003371/ 030 024 210 014 343 023 020 210
004001/ 014 371 020 003 210 014 351 012
004011/ 203 332 012 232 312 012 220 362
004021/ 012 215 372 003 203 334 003 232
004031/ 314 003 220 364 003 215 374 022
004041/ 003 330 022 032 310 022 020 360
004051/ 022 015 370 011 116 333 005 011
004061/ 373 004 011 363 102 370 103 371
004071/ 104 372 105 373 110 374 114 375
004101/ 115 376 123 376 120 376 101 377
004111/ 130 121 124 073 003 114 117 103
004121/ 036 006 104 120 117 257 002 114
004131/ 104 117 307 002 104 114 120 061
004141/ 005 105 104 124 346 002 123 102
004151/ 120 105 003 103 102 120 273 003
004161/ 104 120 123 100 006 103 120 131
004171/ 210 020 124 122 116 275 020 120
004201/ 122 107 142 021 120 124 114 000
004211/ 000 360 114 104 110 000 340 003 015
004221/ 003 077 215 017 026 177 010 014
004231/ 024 166 115 126 011 076 111 016
004241/ 022 074 104 003 022 075 301 004
004251/ 004 207 301 004 003 217 323 025
004261/ 002 227 323 002 002 237 316 004
004271/ 001 247 330 022 001 257 317 022
004301/ 001 267 303 015 020 277 001 104
004311/ 011 306 001 103 011 316 023 125
004321/ 011 326 023 102 011 336 016 104
004331/ 011 346 030 122 011 356 017 122
004341/ 011 366 003 120 011 376 022 014
004351/ 003 007 022 022 003 017 022 001
004361/ 014 027 022 001 022 037 012 215
004371/ 020 303 012 216 003 322 012 216
005001/ 032 302 012 220 005 352 012 220
005011/ 017 342 003 216 003 324 003 216
005021/ 032 304 003 220 005 354 003 220
005031/ 017 344 022 005 024 311 022 016
005041/ 003 320 022 016 032 300 022 020
005051/ 005 350 022 020 017 340 017 125
005061/ 024 323 014 204 001 072 120 017
005071/ 020 371 023 224 001 062 114 230
005101/ 011 071 104 001 104 071 111 016
005111/ 030 073 104 003 130 073 003 015
005121/ 001 057 023 024 003 067 004 001
005131/ 001 047 016 017 020 000 122 023
005141/ 024 377
```

4 LETTER TABLE

2 LETTER TABLE

ARGUMENT TABLE

COMMAND TABLE (SEE LIST BELOW)

XQT LOC DPO LDO DLP EDT
SBP CBP DPS CPY TRN PRG
PTL LDH

120 124 114 000 (PTL
360 114 104 110 000 340 003 015

3 LETTER TABLE

COM

XQT — 003 073
LOC — 006 036
DPO — 002 257
LDO — 002 307
DLP — 006 061
EDT — 002 346
SBP —
CBP —
DPS —
CPY —
TRN —
PRG —
PTL — 360 000
LDH — 340 000

CØNT.ØF MØNITØR
FØLLØWING RØUTINES DEALS WITH SYMBØLIC INP./ØUTP.

```
005143/ 176 MØV   A M            RST?                RST TEST FØR
005144/ 376 CPI       377        -                   LØAD SYMBØLIC
005146/ 302 JNZ       005174     NØ,GØ TØ ARG I/P
005151/ 041 LXI   H   010024     YES,SET H,L TØ RST BUFFER
005154/ 315 CALL      001241     I/P BYTE
005157/ 176 MØV   A M            MASK INTØ 377
005160/ 366 ØRI       307
005162/ 137 MØV   E A            -
005163/ 303 JMP       005354     CØNTINUE IN ARG.I/P
005166/ 315 CALL      000314     CALL 3 LETTER MATCH    LØAD SYMB.
005171/ 043 INX   H              LØAD MASKED INSTR.     RØUTINES(CØNT)
005172/ 136 MØV   E M            CØDE INTØ E
005173/ 053 DCX   H              GØ TØ FIRST BYTE
005174/ 053 DCX   H
005175/ 053 DCX   H
005176/ 053 DCX   H              -
005177/ 176 MØV   A M            MASK,RØTATE,TØ GET ARG.BITS
005200/ 346 NDI       300
005202/ 007 RLC
005203/ 007 RLC                  -
```

```
005204/ 043 INX   H              GØ TØ SEC BYTE
005205/ 312 JZ        005302     NØ ARG,GØ TØ INP.DATA
005210/ 075 DCR   A              1 ARG.DEST.,GØ TØ I/P
005211/ 312 JZ        005271     -
005214/ 075 DCR   A              2 ARG.,GØ TØ I/P
005215/ 312 JZ        005255     -
005220/ 315 CALL      005371     ASUME 1 ARG.SØURCE
005223/ 243 NDA   E              MASK ARG INTØ INSTR.BYTE
005224/ 137 MØV   E A
005225/ 303 JMP       005274     GØ TØ DATA I/P
005230/ 043 INX   H                                  DEST.SUBRØUTINE
005231/ 315 CALL      005371     I/P ARG.
005234/ 107 MØV   B A            EXCEPTIØN TEST
005235/ 176 MØV   A M
005236/ 346 NDI       100
005240/ 170 MØV   A B
005241/ 312 JZ        005246     -
005244/ 366 ØRI       001
005246/ 007 RLC                  RØTATE,MASK INTØ BYTE,
005247/ 007 RLC                  STØRE IN E
005250/ 007 RLC
005251/ 243 NDA   E
005252/ 137 MØV   E A            -
005253/ 053 DCX   H              SET H,L TØ BYTE WITH DATA CØDE
005254/ 311 RET
005255/ 315 CALL      005230     I/P DEST.ARG.          I/P DEST.,
005260/ 315 CALL      003277     WAIT FØR NØNALPHABET.   SØURCE ARG
005263/ 332 JC        005260     -
005266/ 303 JMP       005220     I/P SØURCE
005271/ 315 CALL      005230     I/P DEST.            I/P DEST.ARG.
005274/ 315 CALL      003277     WAIT FØR NØNALPHABET.
005277/ 332 JC        005274     -
005302/ 176 MØV   A M            I/P BYTE WITH DATA CØDE  I/P DATA
005303/ 041 LXI   H   010023     SET H,L  TØ DATA I/P  BUFFER
005306/ 346 NDI       300        MASK
005310/ 312 JZ        005354     NØ DATA JUMP
005313/ 007 RLC                  1 DATA JUMP
005314/ 322 JNC       005362     -
005317/ 315 CALL      005335     ASSUME 2 DATA  BYTES
005322/ 162 MØV   M D            I/P M.S.BYTE
005323/ 315 CALL      002074     -
005326/ 161 MØV   M C            I/P L.S.BYTE
005327/ 315 CAL       002074     -
005332/ 303 JMP       006041     GØ TØ NEXT LINE
005335/ 315 CALL      001241     I/P BYTE TØ BUFFER   2 DATA I/P
005340/ 116 MØV   C M            MØVE TØ C            SUBR.
005341/ 315 CALL      001241     I/P BYTE TØ D
005344/ 126 MØV   D M            -
005345/ 315 CALL      002002     SET H,L TØ CLP
005350/ 163 MØV   M E            MØVE INSTR.BYTE TØ MEM.
005351/ 303 JMP       002074     INCR.CLP,RET
005354/ 315 CALL      005345     NØ DATA,MØVE INSTR.
005357/ 303 JMP       006041     BYTE TØ MEM.,GØ TØ NEXT LINE
005362/ 315 CALL      005341     1 DATA BYTE
005365/ 112 MØV   C D
005366/ 303 JMP       005326     -
005371/ 315 CALL      003277     WAIT FØR             MATCH ARG.
005374/ 322 JNC       005371     ALPHABETICAL        LETTER
005377/ 305 PUSH  B              SAVE B,C
006000/ 001 LXI   B   004065     SET TØ ARG TABLE
006003/ 315 CALL      006012     SWEEP TABLE TØ FIND CØDE
006006/ 003 INX   B              GØ TØ ARG CØDE BYTE
006007/ 012 LDAX  B              MØVE TØ ACC
006010/ 301 PØP   B              RESTØRE B,C
006011/ 311 RET
006012/ 325 PUSH  D              SAVE D,E             SWEEP ARG
006013/ 036 MVI   E   012        SET CØUNT            TABTE SUBR.
006015/ 127 MØV   D A            SAVE ACC IN D
006015/ 012 LDAX  B              CØMPARE BYTES
006017/ 272 CMP   D              -
006020/ 302 JNZ       006025     NØ GØØD,JUMP
```

```
006023/ 321 PØP   D              FØUND,RESTØRE D,E
006024/ 311 RET
006025/ 035 DCR   E
006026/ 312 JZ        006067     END ØF TABLE,ERRØR
006031/ 003 INX   B              GØ TØ NEXT ARG.
006032/ 003 INX   B              -
006033/ 303 JMP       006015     LØØP
006036/ 315 CALL      002251     LØC                PRINTING CLP/
006041/ 315 CALL      001355     CR/LF
006044/ 041 LXI   H   010000     Ø/P L.S.BYTE ØF CLP AND /
006047/ 315 CALL      001323
006052/ 315 CALL      001372     -
006055/ 361 PØP   PSW            RESET STACK
006056/ 303 JMP       000133     GØ TØ BEG.ØF LØAD SYMBØLIC
006061/ 315 CALL      002025     Ø/P CLP
006064/ 303 JMP       006041     -
006067/ 361 PØP   PSW            RESET STACK          ERRØR RØUT.
006070/ 076 MVI   A   077        Ø/P ?
006072/ 315 CALL      001133     -
006075/ 303 JMP       000133     GØ TØ BEG.PF LØAD SYMB.
006100/ 315 CALL      002235     I/P RANGE          DPS RØUTINE,
006103/ 227 SUB   A              CLEAR EXCEPT.FLAG    MAIN PRØGR.
006104/ 062 STA       010011     -
006107/ 315 CALL      002055     Ø/P CLP/
006112/ 315 CALL      002066     Ø/P MEM.AT CLP
006115/ 315 CALL      007062     MATCH TEST PLUS RST TEST
006120/ 315 CALL      007310     EXCEPTIØN TEST
006123/ 315 CALL      001313     Ø/P BLANC
006126/ 006 MVI   B   006        SET MNENØNIC FIELD
006130/ 315 CALL      007010     Ø/P MNEM.
006133/ 076 MVI   A   040        FINISH MNEN.FIELD
006135/ 315 CALL      003263     WITH BLANC
006140/ 006 MVI   B   003        SET ARG.FIELD
006142/ 315 CALL      007107     Ø/P ARG.
006145/ 315 CALL      002135     PRINTING FINISHED?IF NØ,INCR.CLP
006150/ 072 LDA       010026     DATA BYTE TEST
006153/ 247 NDA   A              -
006154/ 312 JZ        006103
006157/ 365 PUSH  PSW            SAVE ACC             DATA Ø/P
006160/ 076 MVI   A   040        FINISH ARG.FIELD
006162/ 315 CALL      003263     WITH BLANC
006165/ 361 PØP   PSW            RESTØRE ACC
006166/ 075 DCR   A              1,2 DATA BYTES TEST
006167/ 312 JZ        006217     -
006172/ 315 CALL      002074     INCR CLP
006175/ 315 CALL      001323     Ø/P MEM.AT CLP
006200/ 315 CALL      002106     DCR CLP
006203/ 315 CALL      001326     Ø/P MEM AT CLP
006206/ 315 CALL      002074     INCR CLP
006211/ 315 CALL      002135     FINISHED?INCR CLP
006214/ 303 JMP       006103     NEW LINE
006217/ 315 CALL      001313     Ø/P BLANC
006222/ 315 CALL      001326     Ø/P MEM.AT CLP
006225/ 303 JMP       006211     NEW LINE
006230/ 021 LXI   D   004010     SET D,E TØ BEG.ØF 2 BYTE TAB.  MATCH
006233/ 076 MVI   A   002        SET TABLE FLAG(T.F.)          TEST
006235/ 062 STA       010030     -
006240/ 107 MØV   B A            SAVE ACC
006241/ 315 CALL      006333     ARG TEST
006244/ 170 MØV   A B            RESTØRE ACC
006245/ 203 ADD   E              INCR.D,E
006246/ 137 MØV   E A
006247/ 322 JNC       006253
006252/ 024 INR   D
006253/ 032 LDAX  D              LØAD CØDE INTØ ACC
006254/ 271 CMP   C              CØMPARE WITH MEM.
006255/ 310 RZ
006256/ 315 CALL      006265     END ØF TABLE?
006261/ 170 MØV   A B            LØAD ACC WITH T.F
006262/ 303 JMP       006235     LØØP
006265/ 376 CPI       363        2 LETTER END       END ØF TABLE TEST
```

```
006267/ 312 JZ          006325      -
006272/ 376 CPI         351         4 LETTER END
006274/ 312 JZ          006320      -
006277/ 376 CPI         377         3 LETTER END
006301/ 312 JZ          006313      -
006304/ 376 CPI         343         4 LETTER SECØUND HALF
006306/ 312 JZ          003313      ERRØR,CØDE NØT FØUND
006311/ 023 INX   D
006312/ 311 RET
006313/ 004 INR   B                 SET T.F TØ 4
006314/ 021 LXI   D     003321      D,E TØ BE. ØF 4 LET.TAB.
006317/ 311 RET
006320/ 005 DCR   B                 SET T.F.TØ 3
006321/ 021 LXI   D     004217      SET D,E TØ BEG ØF 3 LET TAB
006324/ 311 RET
006325/ 004 INR   B                 SET T.F.TØ 4
006326/ 004 INR   B                 -
006327/ 021 LXI   D     003376      SET D,E TØ SEC.HALF ØF 4 LET TAB
006332/ 311 RET
006333/ 032 LDAX  D                 1 LETTER TØ ACC     ARG.TEST
006334/ 007 RLC                     ARG FLAG
006335/ 007 RLC                     INTØ 010027
006336/ 346 NDI         003
006340/ 062 STA         010027      -
006343/ 302 JNZ         006350
006346/ 116 MØV   C M               NØ ARG,C HØLDS INSTR.
006347/ 311 RET
006350/ 075 DCR   A                 1 ØR 2 ARG
006351/ 302 JNZ         006361
006354/ 176 MØV   A M               1 ARG.DEST.
006355/ 366 ØRI         070        MØDIFY INSTR.
006357/ 117 MØV   C A               -
006360/ 311 RET
006361/ 075 DCR   A                 2 ARG ØR 1 ARG.SØURCE
006362/ 302 JNZ         006372
006365/ 176 MØV   A M               2 ARG.MØDIFY INSTR
006366/ 366 ØRI         077
006370/ 117 MØV   C A               -
006371/ 311 RET
006372/ 176 MØV   A M               1 ARG SØURCE
006373/ 366 ØRI         007
006375/ 117 MØV   C A               -
006376/ 311 RET
006377/ 072 LDA         010030      T.F. TØ ACC       GØ TØ 1 LETTER SUBR
007002/ 033 DCX   D                 LØØP UNTILL D,E
007003/ 075 DCR   A                 PØINTS TØ  1 LETTER
007004/ 302 JNZ         007002
007007/ 311 RET                     -
007010/ 315 CALL        006377      GØ TØ FIRST LETTER    PRINT
007013/ 072 LDA         010030      T.F.INTØ C          MNEM.FIELD
007016/ 117 MØV   C A               -
007017/ 315 CALL        007050      PRINT FIRST LETTER
007022/ 023 INX   D                 SET CØNDIT.FØR SEC.LETTER
007023/ 015 DCR   C                 -
007024/ 032 LDAX  D                 GENERATE DATA FLAG
007025/ 346 NDI         300         AND STØRE AT 010026
007027/ 007 RLC
007030/ 007 RLC
007031/ 062 STA         010026      -
007034/ 315 CALL        007050      PRINT SEC LETTER
007037/ 015 DCR   C                 END?
007040/ 310 RZ                      -
007041/ 023 INX   D                 NØ,PRINT THE REST
007042/ 315 CALL        007050
007045/ 303 JMP         007037      -
007050/ 032 LDAX  D                 LØAD  LET.TØ ACC    PRINT RØUT.
007051/ 346 NDI         077         CØNVERT TØ ASCII
007053/ 366 ØRI         100         -
007055/ 315 CALL        001133      O/P
007060/ 005 DCR   B                 FIELD CØUNT
007061/ 311 RET
```

```
007062/ 176 MØV   A M          CHECK                RST TEST
007063/ 346 NDI       307      FØR RST
007065/ 376 CPI       307      -
007067/ 302 JNZ       006230   NØ
007072/ 021 LXI   D   005142   YES,SET D,E AND TF
007075/ 076 MVI   A   003      -
007077/ 062 STA       010030
007102/ 037 RAR                SET ARG FLAG
007103/ 062 STA       010027   -
007106/ 311 RET
007107/ 072 LDA       010027   TEST FØR ARG.        PRINT ARG.
007112/ 247 NDA   A            -                    RØUTINE
007113/ 310 RZ                 NØ
007114/ 075 DCR   A            1 ARG DEST.
007115/ 312 JZ        007144   -
007120/ 075 DCR   A            2 ARG
007121/ 312 JZ        007243   -
007124/ 176 MØV   A M          ASSUME 1 ARG SØURCE
007125/ 366 ØRI       370
007127/ 305 PUSH  B            SAVE B,C
007130/ 001 LXI   B   004066   B,C PØINTS TØ BEG.ØF ARG TAB.
007133/ 315 CALL      006012   SWEEP ARG.TAB.
007136/ 013 DCX   B            LØAD ACC WITH LETTER IN ARG.T.
007137/ 012 LDAX  B            -
007140/ 301 PØP   B            RESTØRE B,C
007141/ 303 JMP       007055   PRINT CØNT ØF ACC
007144/ 176 MØV   A M          1 ARG DEST.
007145/ 346 NDI       307      RST?
007147/ 376 CPI       307      -
007151/ 302 JNZ       007176   NØ
007154/ 076 MVI   A   040      YES,FINISH ARG.FIELD
007156/ 006 MVI   B   004      WITH BLANC
007160/ 315 CALL      003263   -
007163/ 176 MØV   A M          INSTR INRØ ACC
007164/ 346 NDI       070      CØNVERT  TØ ØCT NØ.AND
007166/ 041 LXI   H   010027   SEND TØ MEM.
007171/ 167 MØV   M A          -
007172/ 315 CALL      001326   PRINT ØCTAL BYTE
007175/ 311 RET
007176/ 176 MØV   A M          SP TEST
007177/ 346 NDI       365
007201/ 376 CPI       061      -
007203/ 312 JZ        007255   PRINT SP
007206/ 346 NDI       361      PSW TEST
007210/ 376 CPI       361
007212/ 312 JZ        007270   PRINT PSW
007215/ 116 MØV   C M          EXCEPTIØN TEST
007216/ 072 LDA       010011
007221/ 376 CPI       367      -
007223/ 302 JNZ       007230   NØ EXCEP.
007226/ 241 NDA   C            YES EXEPT.,DECR.ARG.CØDE BY 1
007227/ 117 MØV   C A          -
007230/ 076 MVI   A   070      GENERATING ØF
007232/ 241 NDA   C            ARG.CØDE
007233/ 017 RRC
007234/ 017 RRC
007235/ 017 RRC
007236/ 366 ØRI       370      -
007240/ 303 JMP       007127   GØ TØ PRINT
007243/ 315 CALL      007176   DEST.                2 ARG.PRINT
007246/ 315 CALL      001313   PRINT BLANC
007251/ 005 DCR   B            FILD CØUNT
007252/ 303 JMP       007124   SØURCE
007255/ 076 MVI   A   123                          PRINT SP
007257/ 315 CALL      001133
007262/ 076 MVI   A   120
007264/ 315 CALL      001133
007267/ 311 RET                -
007270/ 076 MVI   A   120                          PRINT PSW
007272/ 315 CALL      001133
007275/ 076 MVI   A   123
```

```
007277/ 315 CALL      001133
007302/ 076 MVI   A   127
007304/ 315 CALL      001133
007307/ 311 RET                    -
007310/ 306 ADI       305          EXCEPT.              EXCEPT.
007312/ 312 JZ        007325       SUSPECT             TEST
007315/ 074 INR   A
007316/ 312 JZ        007325
007321/ 074 INR   A                -
007322/ 302 JNZ       007350       NØ EXEPT.,HLT TEST
007325/ 176 MØV   A M              NØ EXCEPT.
007326/ 376 CPI       072
007330/ 310 RZ                     -
007331/ 346 NDI       010          NØ EXC.
007333/ 310 RZ                     -
007334/ 076 MVI   A   367          YES EXC.
007336/ 062 STA       010011       SET EXC.FLAG
007341/ 072 LDA       010030       INCR.D,E TØ
007344/ 203 ADD   E                NEXT INSTR.IN TABLE
007345/ 137 MØV   E A
007346/ 023 INX   D
007347/ 311 RET                    -
007350/ 176 MØV   A M              HLT TEST
007351/ 376 CPI       166
007353/ 300 RNZ                    -
007354/ 227 SUB   A                HLT,SET ARG.FLAG
007355/ 062 STA       010027       AND INCR.D,E TØ NEXT INSTR.
007360/ 303 JMP       007341       -
020000/ 315 CALL      002227       FØRMAT         CPY RØUT.
020003/ 072 LDA       010016       PRØGRAMMER?
020006/ 376 CPI       001          -
020010/ 302 JNZ       020051       NØ
020013/ 072 LDA       010000       YES;L.LIMIT TØ ACC
020016/ 127 MØV   D A              L.LIM.TØ D
020017/ 072 LDA       010002       U.LIMIT TØ ACC
020022/ 222 SUB   D                U.L-L.L.TØ ACC
020023/ 117 MØV   C A              (U.L.-L.L.)TØ C
020024/ 052 LHLD      010004       NEW RANGE TØ H,L
020027/ 172 MØV   A D              L.L TØ ACC
020030/ 323 ØUT       020          DATA TØ ACC
020032/ 333 IN        002          -
020034/ 167 MØV   M A              DATA TØ MEM.
020035/ 043 INX   H                H,L = H,L PLUS 1
020036/ 024 INR   D                L.L. = L.L. PLUS 1
020037/ 015 DCR   C                CØUNT
020040/ 076 MVI   A   377          TEST
020042/ 271 CMP   C                -
020043/ 312 JZ        000000       FINISHED
020046/ 303 JMP       020027       REPEAT
020051/ 052 LHLD      010004       NØ PRG.,NEW RANGE TØ H,L
020054/ 072 LDA       010000       (L.L.-N.R.)TØ B,C
020057/ 225 SUB   L
020060/ 117 MØV   C A
020061/ 072 LDA       010001
020064/ 234 SBB   H
020065/ 107 MØV   B A              -
020066/ 311 RET
020067/ 322 JNC       020104       (L.L.-N.R.)>0?
020072/ 171 MØV   A C              MØVES UP;TWØS CØMPL.ØF(L.L.-N.R)
020073/ 057 CMA                    INTØ B,C
020074/ 117 MØV   C A
020075/ 170 MØV   A B
020076/ 057 CMA
020077/ 107 MØV   B A
020100/ 003 INX   B                -
020101/ 303 JMP       020146
020104/ 041 LXI   H   010000       MØVES DØWN
020107/ 176 MØV   A M              SUBTRACT (L.L.-N.R.)FRØM L.L.
020110/ 221 SUB   C
020111/ 137 MØV   E A
020112/ 054 INR   L
```

```
020113/ 176 MØV   A M
020114/ 230 SBB   B
020115/ 127 MØV   D A                -
020116/ 052 LHLD      010000         CØPY DATA
020121/ 176 MØV   A M                TØ NEW LØC.
020122/ 022 STAX  D
020123/ 043 INX   H
020124/ 023 INX   D
020125/ 072 LDA       010002
020130/ 275 CMP   L
020131/ 302 JNZ       020121
020134/ 176 MØV   A M
020135/ 022 STAX  D
020136/ 072 LDA       010003
020141/ 274 CMP   H
020142/ 302 JNZ       020121
020145/ 311 RET                      -
020146/ 041 LXI   H   010002         MØVES UP;(L.L.-N.R.)PLUS
020151/ 176 MØV   A M                U.L. INTØ D,E
020152/ 201 ADD   C
020153/ 137 MØV   E A
020154/ 054 INR   L
020155/ 176 MØV   A M
020156/ 210 ADC   B
020157/ 127 MØV   D A                -
020160/ 052 LHLD      010002         CØPY DATA TØ NEW LØC.
020163/ 176 MØV   A M
020164/ 022 STAX  D
020165/ 053 DCX   H
020166/ 033 DCX   D
020167/ 072 LDA       010000
020172/ 275 CMP   L
020173/ 302 JNZ       020163
020176/ 176 MØV   A M
020177/ 022 STAX  D
020200/ 072 LDA       010001
020203/ 274 CMP   H
020204/ 302 JNZ       020163
020207/ 311 RET                      -
020210/ 315 CALL      020000         CPY RØUT.STARTS HERE
020213/ 315 CALL      020067
020216/ 307 RST       000            -
------
020217/ 042 052 062 072 071          TABLE CØNTAINING CØDES FØR
------                               SHLD LHLD STA LDA LXI

020224/ 052 LHLD      010000         L.L.INTØ H,L          TRN RØUT.
020227/ 315 CALL      020246         CALCULATIØN
020232/ 370 RM                       -
020233/ 052 LHLD      010002         U.L.INTØ H,L
020236/ 315 CALL      020246         CALCULATIØN
020241/ 360 RP                       -
020242/ 341 PØP   H                  MØVE UP STACK
020243/ 303 JMP       021024         CHANGE THE ADDRESS
020246/ 353 XCHG                     CALCULAT.;SUBTRACT L.L.,U.L.,
020247/ 052 LHLD      010023         FRØM JMP ADDRESS
020252/ 043 INX   H
020253/ 176 MØV   A M
020254/ 223 SUB   E
020255/ 043 INX   H
020256/ 176 MØV   A M
020257/ 232 SBB   D
020260/ 311 RET                      -
020261/ 052 LHLD      010012         MMM PLUS (U.L.-L.L.)    END ØF
020264/ 353 XCHG                     INTØ D,E                PRØG?
020265/ 052 LHLD      010023         PØINTER INTØ H,L
020270/ 175 MØV   A L                PØINTER-/MMM-(U.L.-L.L.)/>0?
020271/ 223 SUB   E
020272/ 174 MØV   A H
020273/ 232 SBB   D
020274/ 311 RET                      -
```

```
020275/ 315 CALL       002221      TRN RØUT STARTS HERE
020300/ 315 CALL       020051      UP ØR DØWN?
020303/ 322 JNC        020324      MØVES UP;
020306/ 171 MØV    A C             TWØS CØMPL.ØF L.L.-N.R
020307/ 057 CMA
020310/ 117 MØV    C A
020311/ 170 MØV    A B
020312/ 057 CMA
020313/ 107 MØV    B A
020314/ 003 INX    B               -
020315/ 227 SUB    A               MØVES UP FLAG
020316/ 062 STA        010014      -
020321/ 303 JMP        020331
020324/ 076 MVI    A   001          MØVES DØWN
020326/ 062 STA        010014      -
020331/ 315 CALL       021103      CALCULATE MMM PLUS (U.L.-L.L.)
020334/ 042 SHLD       010023      PØINTER INTØ H,L
020337/ 305 PUSH   B               SAVE B
020340/ 315 CALL       006230      MATCH RØUT.
020343/ 315 CALL       007310      EXCEPTIØN RØUT.
020346/ 301 PØP    B               RESTØRE B
020347/ 032 LDAX   D               CØDE TØ 010016
020350/ 062 STA        010016      -
020353/ 315 CALL       006377      FIRST LETTER SUBR.
020356/ 034 INR    E               SECØUND LETTER
020357/ 032 LDAX   D               DATA BYTE TEST
020360/ 036 MVI    E   300
020362/ 243 NDA    E               -
020363/ 376 CPI        000         NØ DATA BYTE
020365/ 312 JZ         021100      -
020370/ 376 CPI        100         1 DATA BYTE
020372/ 312 JZ         021077      -
020375/ 072 LDA        010016      ASSUME 2 DATA BYTES
021000/ 041 LXI    H   020217      IS IT SHLD,LHLD,STA,LDA,LXI?
021003/ 036 MVI    E   005
021005/ 276 CMP    M               -
021006/ 312 JZ         021073      YES,NØ CHANGE
021011/ 054 INR    L
021012/ 035 DCR    E
021013/ 302 JNZ        021005
021016/ 315 CALL       020224      ØUT ØF RANGE?
021021/ 303 JMP        021073      YES,NØ CHANGE
021024/ 052 LHLD       010023      CHANGE
021027/ 043 INX    H               -
021030/ 072 LDA        010014      MØVES DØWN?
021033/ 247 NDA    A
021034/ 302 JNZ        021061      -
021037/ 176 MØV    A M             MØVES UP;ADD (L.L.-N.R.)
021040/ 201 ADD    C               TØ ADDR.ØF JUMP
021041/ 167 MØV    M A
021042/ 043 INX    H
021043/ 176 MØV    A M
021044/ 210 ADC    B
021045/ 167 MØV    M A
021046/ 043 INX    H               -
021047/ 042 SHLD       010023      INCREMENTED PØINTER TØ 010023
021052/ 315 CALL       020261      END ØF PRØG?
021055/ 372 JM         020334      NØ,NEW LINE
021060/ 307 RST        000         YES
021061/ 176 MØV    A M             MØVES DØWN;SUBTRACT
021062/ 221 SUB    C               (L.L.-N.R.)FRØM ADDR.
021063/ 167 MØV    M A
021064/ 043 INX    H
021065/ 176 MØV    A M
021066/ 230 SBB    B
021067/ 167 MØV    M A             -
021070/ 303 JMP        021046      NEW LINE
021073/ 052 LHLD       010023      NØ CHANGE
021076/ 043 INX    H
021077/ 043 INX    H
021100/ 303 JMP        021046      -
```

```
021103/ 305 PUSH  B              SAVE B,C
021104/ 052 LHLD      010002     U.L.-L.L.
021107/ 353 XCHG
021110/ 052 LHLD      010000
021113/ 173 MØV   A E
021114/ 225 SUB   L
021115/ 137 MØV   E A
021116/ 172 MØV   A D
021117/ 234 SBB   H
021120/ 127 MØV   D A            -
021121/ 052 LHLD      010006     MMM PLUS (U.L.-L.L.)
021124/ 173 MØV   A E            INTØ 010012
021125/ 205 ADD   L
021126/ 157 MØV   L A
021127/ 172 MØV   A D
021130/ 214 ADC   H
021131/ 147 MØV   H A
021132/ 042 SHLD      010012     -
021135/ 052 LHLD      010006
021140/ 301 PØP   B              RESTØRE B,C
021141/ 311 RET
021142/ 315 CALL      002235     I/P L.L.,U.L.       PRG.RØUTIN
021145/ 315 CALL      001355     CR/LF
021150/ 076 MVI   A   045        Ø/P %
021152/ 315 CALL      001133     -
021155/ 315 CALL      001001     I/P LETTER
021160/ 007 RLC                  SHIFT,STØRE
021161/ 007 RLC
021162/ 137 MØV   E A            -
021163/ 052 LHLD      010000     L.L.INTØ H,L
021166/ 175 MØV   A L            CØMPARE RØM WITH MEM.
021167/ 323 ØUT       020
021171/ 333 IN        002
021173/ 276 CMP   M              -
021174/ 304 CNZ       021225     PRØGR.IF NØT EQUAL
021177/ 072 LDA       010002     END ØF PRØGRAMM?
021202/ 275 CMP   L
021203/ 302 JNZ       021221
021206/ 072 LDA       010003
021211/ 274 CMP   H
021212/ 302 JNZ       021221     -
021215/ 315 CALL      021225     YES,PRØGR.LAST
021220/ 307 RST       000        -
021221/ 043 INX   H              NEW BYTE
021222/ 303 JMP       021321     -
021225/ 006 MVI   E   001        PRØGR.4 TIMES LØNGER    PRØGRAMMING
021227/ 315 CALL      021252
021232/ 170 MØV   A B
021233/ 007 RLC
021234/ 007 RLC
021235/ 107 MØV   B A
021236/ 315 CALL      021252
021241/ 005 DCR   B
021242/ 302 JNZ       021236     -
021245/ 170 MØV   A B            Ø/P NULL CHARACT.
021246/ 315 CALL      001133
021251/ 311 RET                  -
021252/ 176 MØV   A M            PRØGRAM UNTILL RØM
021253/ 057 CMA                  BYTE HØLDS DESIRED
021254/ 323 ØUT       022        DATA;MAX 255 PRG.PULSES
021256/ 076 MVI   A   004
021260/ 323 ØUT       026
021262/ 257 XRA   A
021263/ 323 ØUT       026
021265/ 113 MØV   C E
021266/ 026 MVI   D   350
021270/ 025 DCR   D
021271/ 302 JNZ       021270
021274/ 015 DCR   C
021275/ 302 JNZ       021266
021300/ 333 IN        002
```

```
021302/ 276 CMP    M
021303/ 310 RZ
021304/ 004 INR    B
021305/ 302 JNZ         021252        -
021310/ 315 CALL        002025        ERRØR;Ø/P ADDRESS,?
021313/ 076 MVI    A    077
021315/ 315 CALL        001133
021320/ 307 RST            ØØØ        -
021321/ Ø42 SHLD           Ø1ØØØØ
021324/ 303 JMP            Ø211ЄЄ
```

021327
↓ NOP
377

INTEL or MOTOROLA.

Computer — 100 → 400
MEMORY — 50 → 800
TERMINAL — 150 — 1000
PERIPHERALS — 50 → INF
TIME — INF.

See Oct 76 QST for suppliers.

4.2 Subroutine Index

List of useful subroutines used by MONITOR 80 which are available to the user.

Start Address	Name	Description	Remark
000357	Save	Will save the contents of all registers plus flags in the stack	Uses the location 010014 010015
000373	Restore	Will restore contents of all registers and flags	
001001	TTY In	Will input the character from TTY into A register	Uses 010013
001117	Timing Loop	General purpose delay	
001133	TTY Out	Output the character in A register on TTY	Uses 110013 Check 010010
001241	I/P Byte	Inputs octal byte to memory addressed by H, L	
001313	O/P Space	Outputs blank character on TTY	
001323	O/P Bytes	Outputs the octal contents of memory addressed by H, L onto TTY	
001355	O/P CR/LF	Output carriage Return/Line Feed	
001372	O/P /	Output /	
003260	O/P with Loop	A registers hold ASCII code, B registers hold number of repetitions	

The following is a complete commented listing of the MONITOR 80 software package.

```
000000/ 076 MVI   A   001        IDLE TTY
000002/ 323 ØUT       024        -
000004/ 303 JMP       000100     CØNTINUE ELSWHERE
000007/ 000 NØP
000010/ 345 PUSH  H              SAVE H,L.      RST 010;BRK.PT.EXEC
000011/ 041 LXI   H   010023     SET TØ PRINT BUFFER(PB)
000014/ 167 MØV   M A            ACC TØ PB
000015/ 303 JMP       003160     CØNT ELSWHERE
000020/ 315 CALL      000357     SAVE.          RST 020
000023/ 052 LHLD      010100     INDIRECT JUMP TØ LØC PØINTED
000026/ 351 PCHL                 BY CØNTENT ØF 010100,010101
000027/ 000 NØP
000030/ 315 CALL      000357                    RST 030
000033/ 052 LHLD      010102     IND.JMP.PØINTED BY
000036/ 351 PCHL                 010102,010103
000037/ 000 NØP
000040/ 315 CALL      000357                    RST 040
000043/ 052 LHLD      010104
000046/ 351 PCHL
000047/ 000 NØP
000050/ 315 CALL      000357                    RST 050
000053/ 052 LHLD      010106
000056/ 351 PCHL
000057/ 000 NØP
000060/ 315 CALL      000357                    RST 060
000063/ 052 LHLD      010110
000066/ 351 PCHL
000067/ 000 NØP
000070/ 315 CALL      000357                    RST 070
000073/ 052 LHLD      010112
000076/ 351 PCHL
000077/ 000 NØP
000100/ 323 ØUT       026        DISABLE R.C.,STØP PRØGRAMMER
000102/ 041 LXI   H   010100     SET S.P
000105/ 371 SPHL                 -
000106/ 076 MVI   A   000        CLEAR TTY I/Ø,PRG,
000110/ 062 STA       010010     EXCEPTIØN FLAGS
000113/ 062 STA       010011
000116/ 062 STA       010016     -
000121/ 076 MVI   A   055        Ø/P ------
000123/ 006 MVI   B   006
```

```
000125/ 315 CALL       003260      -
000130/ 315 CALL       001355      CR/LF
000133/ 315 CALL       003277      I/P NØNBLANK,FIRST LETTER
000136/ 312 JZ         000133      -
000141/ 346 NDI        077         MASK
000143/ 117 MØV   C A              STØRE IN C
000144/ 315 CALL       003277      I/P ALPHABETIC,SECØND LETTER
000147/ 322 JNC        006070      -
000152/ 346 NDI        077         MASK
000154/ 127 MØV   D A              STØRE IN D
000155/ 315 CALL       003277      I/P ALPHABETIC,THIRD LETTER
000160/ 322 JNC        000250      -
000163/ 346 NDI        077         MASK
000165/ 137 MØV   E A              STØRE IN E
000166/ 315 CALL       003277      I/P ALPHABETIC,FØURTH LETTER-
000171/ 322 JNC        000221      -
000174/ 315 CALL       003277      I/P NØNALPHABETIC
000177/ 332 JC         006070      -
000202/ 041 LXI   H    003321      SET H,L TØ 4 LET.TABLE.    4 LET.
000205/ 006 MVI   B    013         INSTR CØUNT                DEC.
000207/ 315 CALL       005166      SWEEP 4 LETTER TABLE    CØNTRØLER
000212/ 043 INX   H                NØ MATCH,GØ TØ NEXT
000213/ 302 JNZ        000207      SET ØF BYTES
000216/ 303 JMP        006070      END ØF 4 LET.TABLE,ERRØR
000221/ 041 LXI   H    004111      SET H,L TØ BEG.ØF CØMMAND TABLE
000224/ 006 MVI   B    016         SET UP CØUNT
000226/ 315 CALL       000270      CALL 3 LET.MATCH,IF NØ
000231/ 043 INX   H                MATCH,GØ TØ NEXT BYTE
000232/ 302 JNZ        000226      NØT END ØF TABLE,TRY AGAIN
000235/ 006 MVI   B    065         NØT CØMMAND,SET CØUNT TØ 3 LET INS.T
000237/ 315 CALL       000300      CALL 3 LET INST MATCH
000242/ 302 JNZ        000237      NØT END,TRY AGAIN
000245/ 303 JMP        006070      END ØF TABLE,ERRØR
000250/ 041 LXI   H    004010      SET H,L TØ 2 LET.TABLE
000253/ 006 MVI   B    017         SET UP CØUNT
000255/ 132 MØV   E D              SHIFT
000256/ 121 MØV   D C              -
000257/ 315 CALL       000306      CALL 2 LET.MATCH
000262/ 302 JNZ        000257      NØ MATCH,TRY AGAIN
000265/ 303 JMP        006070      END ØF TABLE,ERRØR
000270/ 315 CALL       000314      CALL 3 LET.MATCH          EXECUTE
000273/ 136 MØV   E M              MATCH FØUND,LØAD          GIVEN
000274/ 043 INX   H                JUMP ADDR.TØ D,E          CØMMAND
000275/ 126 MØV   D M              -
000276/ 353 XCHG                   INDIRECT JMP
000277/ 351 PCHL                   -
000300/ 315 CALL       000314      CALL 3 LET.MATCH
000303/ 303 JMP        005143      MATCH FØUND,CØNT ELSWHERE
000306/ 315 CALL       000323      2 LET MATCH
000311/ 303 JMP        005175      MATCH FØUND,CØNT ELSWHERE
000314/ 315 CALL       000352      MASK,INCREMENT H,L
000317/ 271 CMP   C                THIRD LETTER EQUAL TØ TAB?
000320/ 302 JNZ        000343      NØ,JMP
000323/ 315 CALL       000352      YES,GØ TØ SECØUND LETTER
000326/ 272 CMP   D                SECØUND LETTER EQUAL?
000327/ 302 JNZ        000344      NØ
000332/ 315 CALL       000352      YES,GØ TØ FIRST LETTER
000335/ 273 CMP   E                LAST LETTER EQUAL?
000336/ 302 JNZ        000345      NØ
000341/ 136 MØV   E M              XES,MØVE CØDE TØ E
000342/ 311 RET
000343/ 043 INX   H
000344/ 043 INX   H
000345/ 043 INX   H
000346/ 063 INX   SP               SET SP
000347/ 063 INX   SP               -
000350/ 005 DCR   B                CØUNT INSTR.
000351/ 311 RET
000352/ 076 MVI   A    077                    MASK
```

```
000354/ 246 NDA   M                          SUBRØUTINE
000355/ 043 INX   H
000356/ 311 RET                              -
000357/ 042 SHLD      010014                 SAVE H,L TEMPØRARILY          SAVE
000362/ 343 XTHL                             EXCH.H,L AND SP               SUBR.
000363/ 365 PUSH  PSW                        SAVE REGISTERS
000364/ 305 PUSH  B
000365/ 325 PUSH  D
000366/ 345 PUSH  H
000367/ 052 LHLD      010014                 RESTØRE H,L
000372/ 311 RET
000373/ 341 PØP   H                                                RESTØRE
000374/ 321 PØP   D                                                SUBR.
000375/ 301 PØP   B
000376/ 361 PØP   PSW
000377/ 343 XTHL
001000/ 311 RET                              -
001001/ 315 CALL      000357                 SAVE REG              I/P TTY
001004/ 072 LDA       010010                 TTYI/P?               SUBR.
001007/ 376 CPI       000
001011/ 302 JNZ       000000                 -
001014/ 016 MVI   C   367                    BIT CØUNT
001016/ 074 INA   ✕                          ENABLE R.C.
001017/ 323 ØUT       026                    -
001021/ 333 IN        000                    WAIT FØR START PULSE
001023/ 017 RRC
001024/ 322 JNC       001021                 -
001027/ 076 MVI   A   000                    DISABLE R.C.
001031/ 323 ØUT       026                    -
001033/ 041 LXI   H   001040                 SET TIME CØNSTANT(T.C.)
001036/ 315 CALL      001117                 TIMEØUT 4.05 MSEC
001041/ 051 DAD   H                          DØUBLE T.C.
001042/ 333 IN        000                    I/P BIT
001044/ 057 CMA                              CØMPLEMENT
001045/ 323 ØUT       024                    ECHØ
001047/ 037 RAR                              RØT.INTØ PØSITIØN
001050/ 170 MØV   A B
001051/ 037 RAR
001052/ 107 MØV   B A                        -
001053/ 315 CALL      001117                 TIMEØUT 9.1 MSEC
001056/ 014 INR   C                          CØUNT BIT
001057/ 302 JNZ       001042                 NØT LAST,LØØP
001062/ 076 MVI   A   001                    IDLE TTY
001064/ 323 ØUT       024                    -
001066/ 051 DAD   H                          DØUBLE T.C.
001067/ 315 CALL      001117                 TIMEØUT 18.2 MSEC
001072/ 170 MØV   A B                        MASK PARITY
001073/ 346 NDI       177                    -
001075/ 062 STA       010013                 PUT INTØ TEMP.STØRE
001100/ 315 CALL      000373                 RESTØRE REG.
001103/ 072 LDA       010013                 TEST FØR TAPE LEADER
001106/ 376 CPI       000                    -
001110/ 312 JZ        001001
001113/ 376 CPI       001                    TEST FØR CTRL A
001115/ 300 RNZ
001116/ 307 RST       000                    IF CTRL A,GØ TØ BEG.ØF MØN.
001117/ 345 PUSH  H                                                TIMING
001120/ 044 INR   H                                                L ØØP
001121/ 055 DCR   L
001122/ 302 JNZ       001121
001125/ 045 DCR   H
001126/ 302 JNZ       001121
001131/ 341 PØP   H
001132/ 311 RET
001133/ 315 CALL      000357                 SAVE REG.             TTY Ø/P
001136/ 016 MVI   C   370                    CØUNTER
001140/ 062 STA       010013                 MØVE TØ I/Ø BUFFER
001143/ 072 LDA       010010                 TTY I/P TEST
001146/ 376 CPI       000
001150/ 302 JNZ       000000
001153/ 323 ØUT       024                    START TTY
```

```
001155/ 041 LXI   H   001040      SET T.C.
001160/ 051 DAD   H               DØUBLE T.C.
001161/ 072 LDA       010013      RESTØRE ACC
001164/ 315 CALL      001117      TIMEØUT 9.1 MSEC
001167/ 323 ØUT       024         Ø/P BIT
001171/ 037 RAR                   RØTATE
001172/ 014 INR   C               CØUNT
001173/ 302 JNZ       001164      CØNTINUE ØUTPUTING
001176/ 315 CALL      001117      TIMEØUT 9.1 MSEC
001201/ 076 MVI   A   001         IDLE TTY
001203/ 323 ØUT       024         -
001205/ 051 DAD   H               TIMEØUT 18.2 MSEC
001206/ 315 CALL      001117      -
001211/ 315 CALL      000373      RESTØRE REG
001214/ 311 RET
001215/ 315 CALL      001001                          I/P WITH
001220/ 107 MØV   B A                                 ØCTAL TEST
001221/ 346 NDI       370                             INTØ B
001223/ 376 CPI       060
001225/ 311 RET                   -
001226/ 315 CALL      001001                          I/P WITH
001231/ 376 CPI       177                             RUBØUT
001233/ 300 RNZ
001234/ 076 MVI   A   137         PRINT
001236/ 303 JMP       001133      -
001241/ 315 CALL      000357      SAVE REG.           I/P BYTE
001244/ 315 CALL      001215      I/P ØCT             TØ MEMØRY
001247/ 302 JNZ       001244      -
001252/ 170 MØV   A B             RØTATE,SAVE IN B
001253/ 017 RRC
001254/ 017 RRC
001255/ 346 NDI       300
001257/ 107 MØV   B A             -
001260/ 315 CALL      001226      I/P ASCII WITH RUBØUT TEST
001263/ 312 JZ        001244      -
001266/ 346 NDI       007         MASK,RØTATE,SAVE IN C
001270/ 007 RLC
001271/ 007 RLC
001272/ 007 RLC
001273/ 117 MØV   C A             -
001274/ 315 CALL      001226      I/P ASCII WITH RUBØUT TDST
001277/ 312 JZ        001260      -
001302/ 346 NDI       007         MASK CØMBINE,
001304/ 260 ØRA   B               MØVE TØ MEMØRY
001305/ 261 ØRA   C
001306/ 167 MØV   M A             -
001307/ 315 CALL      000373      RESTØRE REG.
001312/ 311 RET
001313/ 365 PUSH  PSW                                 Ø/P BLANC
001314/ 076 MVI   A   040
001316/ 315 CALL      001133
001321/ 361 PØP   PSW
001322/ 311 RET                   -
001323/ 315 CALL      001313      Ø/P BL              Ø/P MEMØRY
001326/ 176 MØV   A M             Ø/P FIRST DIGIT     BYTE
001327/ 007 RLC
001330/ 007 RLC
001331/ 346 NDI       003
001333/ 315 CALL      001350      -
001336/ 176 MØV   A M             Ø/P SEC.DIGIT
001337/ 017 RRC
001340/ 017 RRC
001341/ 017 RRC
001342/ 315 CALL      001346      -
001345/ 176 MØV   A M             Ø/P LAST DIGIT
001346/ 346 NDI       007
001350/ 366 ØRI       060
001352/ 303 JMP       001133      -
001355/ 365 PUSH  PSW                                 Ø/P CR/LF
001356/ 076 MVI   A   015
001360/ 315 CALL      001133
001363/ 076 MVI   A   012
```

```
001365/ 315 CALL      001133
001370/ 361 PØP   PSW
001371/ 311 RET
001372/ 365 PUSH  PSW                                  Ø/P /(SLASH)
001373/ 076 MVI   A   057
001375/ 315 CALL      001133
002000/ 361 PØP   PSW
002001/ 311 RET
002002/ 052 LHLD      010000     SET TØ CLP.          SET H,L TØ CLP
002005/ 072 LDA       010016     TEST FØR PRG.
002010/ 376 CPI       001
002012/ 300 RNZ
002013/ 175 MØV   A L            YES,PRG.SET TØ PRG.BUFFER
002014/ 041 LXI   H   010007
002017/ 323 ØUT       020        Ø/P ADDRESS
002021/ 333 IN        002        I/P BYTE
002023/ 167 MØV   M A            MØVE TØ BUFFER
002024/ 311 RET
002025/ 315 CALL      002002     SET H,L TØ CLP.      Ø/P CLP
002030/ 041 LXI   H   010001     SET TØ M.S.BYTE ØF CLP
002033/ 302 JNZ       002046     NØT PRG,JUMP
002036/ 076 MVI   A   120        Ø/P P(PRØGRAMMER)
002040/ 315 CALL      001133
002043/ 303 JMP       002051
002046/ 315 CALL      001323     PRINT M.S.BYTE ØF ADDRESS
002051/ 053 DCX   H
002052/ 303 JMP       001326     PRINT L.S.BYTE ØF ADDRESS
002055/ 315 CALL      001355                          Ø/P CLP/
002060/ 315 CALL      002025
002063/ 303 JMP       001372
002066/ 315 CALL      002002                          Ø/P MEMØRY AT CLP
002071/ 303 JMP       001323
002074/ 052 LHLD      010000                          INCREMENT CLP
002077/ 043 INX   H
002100/ 042 SHLD      010000
002103/ 303 JMP       002002
002106/ 052 LHLD      010000                          DECREMENT CLP
002111/ 053 DCX   H
002112/ 042 SHLD      010000
002115/ 303 JMP       002002
002120/ 052 LHLD      010000                          CØMPARE CLP
002123/ 353 XCHG                                      AND UPPER LIMIT
002124/ 052 LHLD      010002
002127/ 172 MØV   A D
002130/ 274 CMP   H
002131/ 300 RNZ
002132/ 173 MØV   A E
002133/ 275 CMP   L
002134/ 311 RET
002135/ 315 CALL      002120     CØMPARE CLP AND U.L.      RANGE
002140/ 302 JNZ       002074     NØT ZERØ,INCR.CLP        CØNTRØL
002143/ 307 RST       000        END ØF PRØGRAMM,GØ TØ BEG.
002144/ 315 CALL      001355                          Ø/P CLP*
002147/ 076 MVI   A   052
002151/ 315 CALL      001133
002154/ 315 CALL      001313    Ø/P BLANC            I/P ADDR.TØ MEM
002157/ 315 CALL      001001     I/P ASCII
002162/ 376 CPI       120        =P?
002164/ 312 JZ        002210     YES SET PRG FLAG
002167/ 001 LXI   B   002204     NØ,SET UP NEXT ADDR.-I/P M.S.BYTE
002172/ 305 PUSH  B
002173/ 315 CALL      000357     SAVE REG.
002176/ 315 CALL      001220     I/P ØCTAL
002201/ 303 JMP       001247
002204/ 053 DCX   H              I/P BYTE TØ MEM.-I/P L.S.BYTE
002205/ 303 JMP       001241
002210/ 076 MVI   A   001        SET PRØGRAMMER FLAG
002212/ 062 STA       010016
002215/ 167 MØV   M A            STØRE P
002216/ 303 JMP       002204     I/P WØRD
002221/ 041 LXI   H   010007     I/P LØC WHERE PRØG.       I/P LIMITS
```

```
002224/ 315 CALL      002144      IS NØW
002227/ 041 LXI   H   010005      I/P NEW LØC.
002232/ 315 CALL      002144      -
002235/ 315 CALL      002251      Ø/P CR/LF*,I/P NEW CLP
002240/ 041 LXI   H   010003      Ø/P BL.,I/P UPPER L.
002243/ 315 CALL      002154      -
002246/ 303 JMP       001355      Ø/P CR/LF,RET
002251/ 041 LXI   H   010001      SET H,L TØ CLP        DPØ
002254/ 303 JMP       002144      Ø/P CR/LF*,I/P CLP
002257/ 315 CALL      002235      I/P UPPER L.,Ø/P CR/LF
002262/ 016 MVI   C   010         SET UP CØUNT
002264/ 315 CALL      002055      Ø/P CLP/
002267/ 315 CALL      002002      SET H,L TØ CLP
002272/ 315 CALL      001323      Ø/P MEMØRY BYTE
002275/ 315 CALL      002135      INCR CLP
002300/ 015 DCR   C               CØUNT
002301/ 302 JNZ       002272      LØØP
002304/ 303 JMP       002262      NEW LINE
002307/ 315 CALL      002235      I/P CLP,UPPER L.,Ø/P CR/LF     LDØ
002312/ 315 CALL      001001      I/P ASCII
002315/ 376 CPI       057         WAIT FØR /
002317/ 302 JNZ       002312      -
002322/ 315 CALL      002002      SET H,L TØ CLP
002325/ 315 CALL      001001      I/P ASCII,TEST
002330/ 376 CPI       015         FØR CR/LF
002332/ 312 JZ        002312      -
002335/ 315 CALL      001241      I/P BYTE TØ MEM.
002340/ 315 CALL      002135      RANGE CØNTRØL (END?)
002343/ 303 JMP       002325      LØØP
002346/ 315 CALL      002251      I/P CLP             ØCTAL
002351/ 315 CALL      002055      Ø/P CLP             EDITØR
002354/ 315 CALL      002002      SET H,L TØ CLP
002357/ 315 CALL      002365      PRØCESS LINE
002362/ 303 JMP       002351      LØØP
002365/ 315 CALL      001001      I/P ASCII           EDITING
002370/ 376 CPI       122         R?                  RØUTINES
002372/ 312 JZ        000000      YES,GØ TØ BEG.
002375/ 376 CPI       052         *?
002377/ 312 JZ        002251      YES,GØ TØ I/P CLP RØUTINE
003002/ 376 CPI       100          @?
003004/ 312 JZ        003073      XES,GØ TØ XQT
003007/ 376 CPI       136         ↑?
003011/ 312 JZ        002106      YES,GØ TØ DECR.CLP
003014/ 376 CPI       040         BLANC?
003016/ 312 JZ        003042      YES,PRINT ØCTAL BYTE
003021/ 315 CALL      001220      TEST FØR ØCTAL
003024/ 300 RNZ                   NØT ØCTAL
003025/ 021 LXI   D   003037      I/P BYTE TØ MEM.
003030/ 325 PUSH  D
003031/ 315 CALL      000357
003034/ 303 JMP       001252
003037/ 303 JMP       003045
003042/ 315 CALL      002066      Ø/P MEM AT CLP
003045/ 315 CALL      001001      I/P ASCII
003050/ 376 CPI       137         BACK ARRØW?
003052/ 302 JNZ       003063      NØ,JUMP
003055/ 315 CALL      001241      YES,I/P ANØTHER BYTE
003060/ 303 JMP       003045      LØØP
003063/ 376 CPI       040         BLANC?
003065/ 312 JZ        002074      YES,INCR.CLP AND RET.
003070/ 303 JMP       002370      NØ.LØØP
003073/ 315 CALL      001355      CR/LF               XQT
003076/ 315 CALL      002251      I/P ADDRES TØ H,L
003101/ 052 LHLD      010000      -
003104/ 351 PCHL                  INDIRECT JUMP
003105/ 315 CALL      003135      CLEAR BPT         SET BRKPØINT
003110/ 041 LXI   H   010020      SET H,L TØ BPT ADDRESS BUFFER
003113/ 315 CALL      002144      I/P BPT ADDR.TØ MEM.
003116/ 052 LHLD      010017      SAVE PRØGR.BYTE
003121/ 176 MØV   A M
003122/ 062 STA       010021      -
```

```
003125/ 066 MVI    M    317          INSERT RST 010 INTØ PRØG.
003127/ 076 MVI    A    001          SET BPT FLAG
003131/ 062 STA         010022
003134/ 307 RST         000          -
003135/ 072 LDA         010022       TEST FØR BPT          CLEAR BPT
003140/ 376 CPI         001
003142/ 300 RNZ                      -
003143/ 052 LHLD        010017       LØAD H,LWITH BPT ADDR.
003146/ 072 LDA         010021       MØV PRØG BYTE TØ ACC
003151/ 167 MØV    M A               MØVE BACK INTØ PRØGRAMM
003152/ 076 MVI    A    000          CLEAR BPT  FLAG
003154/ 062 STA         010022
003157/ 311 RET                      -
003160/ 365 PUSH   PSW               SAVE ACC,FLAGS.       EXEC.BPT
003161/ 315 CALL        001323       Ø/P ACC
003164/ 160 MØV    M B               Ø/P B
003165/ 315 CALL        001323       -
003170/ 161 MØV    M C               Ø/P C
003171/ 315 CALL        001323       -
003174/ 162 MØV    M D               Ø/P D
003175/ 315 CALL        001323       -
003200/ 163 MØV    M E               Ø/P E
003201/ 315 CALL        001323       -
003204/ 301 PØP    B                 RESTØRE FLAGS,ACC TØ B,C
003205/ 321 PØP    D                 RESTØRE H,L TØ D,E
003206/ 162 MØV    M D               Ø/P H
003207/ 315 CALL        001323       -
003212/ 163 MØV    M E               Ø/P L
003213/ 315 CALL        001323       -
003216/ 353 XCHG                     Ø/P MEM.
003217/ 315 CALL        001323       -
003222/ 171 MØV    A C               FLAGS TØ ACC
003223/ 315 CALL        003250       Ø/P CARRY
003226/ 315 CALL        003244       Ø/P PARITY
003231/ 315 CALL        003244       Ø/P CY1(CARRY BETWEEN D3,D4)
003234/ 315 CALL        003244       Ø/P ZERØ
003237/ 171 MØV    A C               Ø/P SIGN
003240/ 315 CALL        003246
003243/ 307 RST         000          -
003244/ 171 MØV    A C                                SHIFT ARØUND
003245/ 017 RRC                                       FLAG BYTE
003246/ 017 RRC
003247/ 117 MØV    C A
003250/ 346 NDI         001
003252/ 315 CALL        001313
003255/ 303 JMP         001346       -
003260/ 315 CALL        001355       CR/LF                 Ø/P WITH
003263/ 315 CALL        001133       Ø/P ASCII             CØUNT LØØP
003266/ 005 DCR    B                 CØUNT
003267/ 310 RZ
003270/ 303 JMP         003263       LØØP
003273/ 315 CALL        003135       CLEAR BPT
003276/ 307 RST         000          -
003277/ 315 CALL        001001       I/P ASCII             I/P NUMERIC
003302/ 376 CPI         040          BLANC?                ALPHABETIC,BLANC
003304/ 305 PUSH   B                 SAVE B,C              TEST
003305/ 107 MØV    B A
003306/ 007 RLC                      RØTATE M.S.BIT INTØ CARRY
003307/ 007 RLC
003310/ 170 MØV    A B
003311/ 301 PØP    B                 RESTØRE B,C
003312/ 311 RET                      -
003313/ 076 MVI    A    077                           ERRØR,Ø/P ?
003315/ 315 CALL        001133                         AND RESTART
003320/ 307 RST         000          ----------
     FØLLØWING ARE VARIØUS LØØK UP TABLES
     (SEE MØNITØR STRUCTURE)

003321/ 003 201 214 014 315 120 025 223
003331/ 010 375 030 003 210 007 353 023
003341/ 210 214 004 042 014 210 214 004
```

```
003351/ 052 123 024 201 030 072 114 004
003361/ 301 030 072 030 003 210 007 353
003371/ 030 024 210 014 343 023 020 210
004001/ 014 371 020 003 210 014 351 012
004011/ 203 332 012 232 312 012 220 362
004021/ 012 215 372 003 203 334 003 232
004031/ 314 003 220 364 003 215 374 022
004041/ 003 330 022 032 310 022 020 360
004051/ 022 015 370 011 116 333 005 011
004061/ 373 004 011 363 102 370 103 371
004071/ 104 372 105 373 110 374 114 375
004101/ 115 376 123 376 120 376 101 377
004111/ 130 121 124 073 003 114 117 103
004121/ 036 006 104 120 117 257 002 114
004131/ 104 117 307 002 104 114 120 061
004141/ 006 105 104 124 346 002 123 102
004151/ 120 105 003 103 102 120 273 003
004161/ 104 120 123 100 006 103 120 131
004171/ 210 020 124 122 116 275 020 120
004201/ 122 107 142 021 000 000 000 000
004211/ 000 000 000 000 000 000 003 015
004221/ 003 077 215 017 026 177 010 014
004231/ 024 166 115 126 011 076 111 016
004241/ 022 074 104 003 022 075 301 004
004251/ 004 207 301 004 003 217 323 025
004261/ 002 227 323 002 002 237 301 016
004271/ 001 247 330 022 001 257 317 022
004301/ 001 267 303 015 020 277 001 104
004311/ 011 306 001 103 011 316 023 125
004321/ 011 326 023 102 011 336 001 116
004331/ 011 346 030 122 011 356 017 122
004341/ 011 366 003 120 011 376 022 014
004351/ 003 007 022 022 003 017 022 001
004361/ 014 027 022 001 022 037 012 215
004371/ 020 303 012 216 003 322 012 216
005001/ 032 302 012 220 005 352 012 220
005011/ 017 342 003 216 003 324 003 216
005021/ 032 304 003 220 005 354 003 220
005031/ 017 344 022 005 024 311 022 016
005041/ 003 320 022 016 032 300 022 020
005051/ 005 350 022 020 017 340 017 125
005061/ 024 323 014 204 001 072 120 017
005071/ 020 371 023 224 001 062 114 230
005101/ 011 071 104 001 104 071 111 016
005111/ 030 073 104 003 130 073 003 015
005121/ 001 057 023 024 003 067 004 001
005131/ 001 047 016 017 020 000 122 023
005141/ 024 377
   --------
     CØNT.ØF MØNITØR
     FØLLØWING RØUTINES DEALS WITH SYMBØLIC INP./ØUTP.
005143/ 176 MØV   A M             RST?                    RST TEST FØR
005144/ 376 CPI       377         -                       LØAD SYMBØLIC
005146/ 302 JNZ       005174      NØ,GØ TØ ARG I/P
005151/ 041 LXI   H   010024      YES,SET H,L TØ RST BUFFER
005154/ 315 CALL      001241      I/P BYTE
005157/ 176 MØV   A M             MASK INTØ 377
005160/ 366 ØRI       307
005162/ 137 MØV   E A             -
005163/ 303 JMP       005354      CØNTINUE IN ARG.I/P
005166/ 315 CALL      000314      CALL 3 LETTER MATCH     LØAD SYMB.
005171/ 043 INX   H               LØAD MASKED INSTR.      RØUTINES(CØNT)
005172/ 136 MØV   E M             CØDE INTØ E
005173/ 053 DCX   H               GØ TØ FIRST BYTE
005174/ 053 DCX   H
005175/ 053 DCX   H
005176/ 053 DCX   H               -
005177/ 176 MØV   A M             MASK,RØTATE,TØ GET ARG.BITS
005200/ 346 NDI       300
005202/ 007 RLC
005203/ 007 RLC                   -
```

```
005204/ 043 INX   H                 GØ TØ SEC BYTE
005205/ 312 JZ        005302        NØ ARG,GØ TØ INP.DATA
005210/ 075 DCR   A                 1 ARG.DEST.,GØ TØ I/P
005211/ 312 JZ        005271        -
005214/ 075 DCR   A                 2 ARG.,GØ TØ I/P
005215/ 312 JZ        005255        -
005220/ 315 CALL      005371        ASUME 1 ARG.SØURCE
005223/ 243 NDA   E                 MASK ARG INTØ INSTR.BYTE
005224/ 137 MØV   E A
005225/ 303 JMP       005274        GØ TØ DATA I/P
005230/ 043 INX   H                                   DEST.SUBRØUTINE
005231/ 315 CALL      005371        I/P ARG.
005234/ 107 MØV   B A               EXCEPTIØN TEST
005235/ 176 MØV   A M
005236/ 346 NDI       100
005240/ 170 MØV   A B
005241/ 312 JZ        005246        -
005244/ 366 ØRI       001
005246/ 007 RLC                     RØTATE,MASK INTØ BYTE,
005247/ 007 RLC                     STØRE IN E
005250/ 007 RLC
005251/ 243 NDA   E
005252/ 137 MØV   E A               -
005253/ 053 DCX   H                 SET H,L TØ BYTE WITH DATA CØDE
005254/ 311 RET
005255/ 315 CALL      005230        I/P DEST.ARG.          I/P DEST.,
005260/ 315 CALL      003277        WAIT FØR NØNALPHABET.   SØURCE ARG
005263/ 332 JC        005260        -
005266/ 303 JMP       005220        I/P SØURCE
005271/ 315 CALL      005230        I/P DEST.              I/P DEST.ARG.
005274/ 315 CALL      003277        WAIT FØR NØNALPHABET.
005277/ 332 JC        005274        -
005302/ 176 MØV   A M               I/P BYTE WITH DATA CØDE  I/P DATA
005303/ 041 LXI   H   010023        SET H,L  TØ DATA I/P  BUFFER
005306/ 346 NDI       300           MASK
005310/ 312 JZ        005354        NØ DATA JUMP
005313/ 007 RLC                     1 DATA JUMP
005314/ 322 JNC       005362        -
005317/ 315 CALL      005335        ASSUME 2 DATA  BYTES
005322/ 162 MØV   M D               I/P M.S.BYTE
005323/ 315 CALL      002074        -
005326/ 161 MØV   M C               I/P L.S.BYTE
005327/ 315 CAL       002074        -
005332/ 303 JMP       006041        GØ TØ NEXT LINE
005335/ 315 CALL      001241        I/P BYTE TØ BUFFER   2 DATA I/P
005340/ 116 MØV   C M               MØVE TØ C             SUBR.
005341/ 315 CALL      001241        I/P BYTE TØ D
005344/ 126 MØV   D M               -
005345/ 315 CALL      002002        SET H,L TØ CLP
005350/ 163 MØV   M E               MØVE INSTR.BYTE TØ MEM.
005351/ 303 JMP       002074        INCR.CLP,RET
005354/ 315 CALL      005345        NØ DATA,MØVE INSTR.
005357/ 303 JMP       006041        BYTE TØ MEM.,GØ TØ NEXT LINE
005362/ 315 CALL      005341        1 DATA BYTE
005365/ 112 MØV   C D
005366/ 303 JMP       005326        -
005371/ 315 CALL      003277        WAIT FØR              MATCH ARG.
005374/ 322 JNC       005371        ALPHABETICAL         LETTER
005377/ 305 PUSH  B                 SAVE B,C
006000/ 001 LXI   B   004065        SET TØ ARG TABLE
006003/ 315 CALL      006012        SWEEP TABLE TØ FIND CØDE
006006/ 003 INX   B                 GØ TØ ARG CØDE BYTE
006007/ 012 LDAX  B                 MØVE TØ ACC
006010/ 301 PØP   B                 RESTØRE B,C
006011/ 311 RET
006012/ 325 PUSH  D                 SAVE D,E              SWEEP ARG
006013/ 036 MVI   E   012           SET CØUNT             TABTE SUBR.
006015/ 127 MØV   D A               SAVE ACC IN D
006016/ 012 LDAX  B                 CØMPARE BYTES
006017/ 272 CMP   D                 -
006020/ 302 JNZ       006025        NØ GØØD,JUMP
```

```
006023/ 321 PØP   D              FØUND,RESTØRE D,E
006024/ 311 RET
006025/ 035 DCR   E
006026/ 312 JZ        006067     END ØF TABLE,ERRØR
006031/ 003 INX   B              GØ TØ NEXT ARG.
006032/ 003 INX   B              -
006033/ 303 JMP       006016     LØØP
006036/ 315 CALL      002251     LØC                PRINTING CLP/
006041/ 315 CALL      001355     CR/LF
006044/ 041 LXI   H   010000     Ø/P L.S.BYTE ØF CLP AND /
006047/ 315 CALL      001323
006052/ 315 CALL      001372     -
006055/ 361 PØP   PSW            RESET STACK
006056/ 303 JMP       000133     GØ TØ BEG.ØF LØAD SYMBØLIC
006061/ 315 CALL      002025     Ø/P CLP
006064/ 303 JMP       006041     -
006067/ 361 PØP   PSW            RESET STACK          ERRØR RØUT.
006070/ 076 MVI   A   077        Ø/P ?
006072/ 315 CALL      001133     -
006075/ 303 JMP       000133     GØ TØ BEG.PF LØAD SYMB.
006100/ 315 CALL      002235     I/P RANGE          DPS RØUTINE,
006103/ 227 SU    A              CLEAR EXCEPT.FLAG    MAIN PRØGR.
006104/ 062 STA       010011     -
006107/ 315 CALL      002055     Ø/P CLP/
006112/ 315 CALL      002066     Ø/P MEM.AT CLP
006115/ 315 CALL      007062     MATCH TEST PLUS RST TEST
006120/ 315 CALL      007310     EXCEPTIØN TEST
006123/ 315 CALL      001313     Ø/P BLANC
006126/ 006 MVI   B   006        SET MNENØNIC FIELD
006130/ 315 CALL      007010     Ø/P MNEM.
006133/ 076 MVI   A   040        FINISH MNEN.FIELD
006135/ 315 CALL      003263     WITH BLANC
006140/ 006 MVI   B   003        SET ARG.FIELD
006142/ 315 CALL      007107     Ø/P ARG.
006145/ 315 CALL      002135     PRINTING FINISHED?IF NØ,INCR.CLP
006150/ 072 LDA       010026     DATA BYTE TEST
006153/ 247 NDA   A              -
006154/ 312 JZ        006103
006157/ 365 PUSH  PSW            SAVE ACC             DATA Ø/P
006160/ 076 MVI   A   040        FINISH ARG.FIELD
006162/ 315 CALL      003263     WITH BLANC
006165/ 361 PØP   PSW            RESTØRE ACC
006166/ 075 DCR   A              1,2 DATA BYTES TEST
006167/ 312 JZ        006217     -
006172/ 315 CALL      002074     INCR CLP
006175/ 315 CALL      001323     Ø/P MEM.AT CLP
006200/ 315 CALL      002106     DCR CLP
006203/ 315 CALL      001326     Ø/P MEM AT CLP
006206/ 315 CALL      002074     INCR CLP
006211/ 315 CALL      002135     FINISHED?INCR CLP
006214/ 303 JMP       006103     NEW LINE
006217/ 315 CALL      001313     Ø/P BLANC
006222/ 315 CALL      001326     Ø/P MEM.AT CLP
006225/ 303 JMP       006211     NEW LINE
006230/ 021 LXI   D   004010     SET D,E TØ BEG.ØF 2 BYTE TAB.  MATCH
006233/ 076 MVI   A   002        SET TABLE FLAG(T.F.)          TEST
006235/ 062 STA       010030     -
006240/ 107 MØV   B A            SAVE ACC
006241/ 315 CALL      006333     ARG TEST
006244/ 170 MØV   A B            RESTØRE ACC
006245/ 203 ADD   E              INCR.D,E
006246/ 137 MØV   E A
006247/ 322 JNC       006253
006252/ 024 INR   D
006253/ 032 LDAX  D              LØAD CØDE INTØ ACC
006254/ 271 CMP   C              CØMPARE WITH MEM.
006255/ 310 RZ
006256/ 315 CALL      006265     END ØF TABLE?
006261/ 170 MØV   A B            LØAD ACC WITH T.F
006262/ 303 JMP       006235     LØØP
006265/ 376 CPI       363        2 LETTER END         END ØF TABLE TEST
```

```
006267/ 312 JZ          006325      -
006272/ 376 CPI         351         4 LETTER END
006274/ 312 JZ          006320      -
006277/ 376 CPI         377         3 LETTER END
006301/ 312 JZ          006313      -
006304/ 376 CPI         343         4 LETTER SECØUND HALF
006306/ 312 JZ          003313      ERRØR,CØDE NØT FØUND
006311/ 023 INX   D
006312/ 311 RET
006313/ 004 INR   B                 SET T.F TØ 4
006314/ 021 LXI   D     003321      D,E TØ BE. ØF 4 LET.TAB.
006317/ 311 RET
006320/ 005 DCR   B                 SET T.F.TØ 3
006321/ 021 LXI   D     004217      SET D,E TØ BEG ØF 3 LET TAB
006324/ 311 RET
006325/ 004 INR   B                 SET T.F.TØ 4
006326/ 004 INR   B                 -
006327/ 021 LXI   D     003376      SET D,E TØ SEC.HALF ØF 4 LET TAB
006332/ 311 RET
006333/ 032 LDAX  D                 1 LETTER TØ ACC      ARG.TEST
006334/ 007 RLC                     ARG FLAG
006335/ 007 RLC                     INTØ 010027
006336/ 346 NDI         003
006340/ 062 STA         010027      -
006343/ 302 JNZ         006350
006346/ 116 MØV   C M               NØ ARG,C HØLDS INSTR.
006347/ 311 RET
006350/ 075 DCR   A                 1 ØR 2 ARG
006351/ 302 JNZ         006361
006354/ 176 MØ    A M               1 ARG.DEST.
006355/ 366 ØRI         070         MØDIFY INSTR.
006357/ 117 MØV   C A               -
006360/ 311 RET
006361/ 075 DCR   A                 2 ARG ØR 1 ARG.SØURCE
006362/ 302 JNZ         006372
006365/ 176 MØV   A M               2 ARG.MØDIFY INSTR
006366/ 366 ØRI         077
006370/ 117 MØV   C A               -
006371/ 311 RET
006372/ 176 MØV   A M               1 ARG SØURCE
006373/ 366 ØRI         007
006375/ 117 MØV   C A               -
006376/ 311 RET
006377/ 072 LDA         010030      T.F. TØ ACC        GØ TØ 1 LETTER SUBR
007002/ 033 DCX   D                 LØØP UNTILL D,E
007003/ 075 DCR   A                 PØINTS TØ  1 LETTER
007004/ 302 JNZ         007002
007007/ 311 RET                     -
007010/ 315 CALL        006377      GØ TØ FIRST LETTER    PRINT
007013/ 072 LDA         010030      T.F.INTØ C          MNEM.FIELD
007016/ 117 MØV   C A               -
007017/ 315 CALL        007050      PRINT FIRST LETTER
007022/ 023 INX   D                 SET CØNDIT.FØR SEC.LETTER
007023/ 015 DCR   C                 -
007024/ 032 LDAX  D                 GENERATE DATA FLAG
007025/ 346 NDI         300         AND STØRE AT 010026
007027/ 007 RLC
007030/ 007 RLC
007031/ 062 STA         010026      -
007034/ 315 CALL        007050      PRINT SEC LETTER
007037/ 015 DCR   C                 END?
007040/ 310 RZ                      -
007041/ 023 INX   D                 NØ,PRINT THE REST
007042/ 315 CALL        007050
007045/ 303 JMP         007037      -
007050/ 032 LDAX  D                 LØAD  LET.TØ ACC    PRINT RØUT.
007051/ 346 NDI         077         CØNVERT TØ ASCII
007053/ 366 ØRI         100         -
007055/ 315 CALL        001133      O/P
007060/ 005 DCR   B                 FIELD CØUNT
007061/ 311 RET
```

```
007062/ 176 MØV   A M          CHECK                  RST TEST
007063/ 346 NDI       307      FØR RST
007065/ 376 CPI       307      -
007067/ 302 JNZ       006230   NØ
007072/ 021 LXI   D   005142   YES,SET D,E AND TF
007075/ 076 MVI   A   003      -
007077/ 062 STA       010030
007102/ 037 RAR                SET ARG FLAG
007103/ 062 STA       010027   -
007106/ 311 RET
007107/ 072 LDA       010027   TEST FØR ARG.     PRINT ARG.
007112/ 247 NDA   A            -                 RØUTINE
007113/ 310 RZ                 NØ
007114/ 075 DCR   A            1 ARG DEST.
007115/ 312 JZ        007144   -
007120/ 075 DCR   A            2 ARG
007121/ 312 JZ        007243   -
007124/ 176 MØV   A M          ASSUME 1 ARG SØURCE
007125/ 366 ØRI       370
007127/ 305 PUSH  B            SAVE B,C
007130/ 001 LXI   B   004066   B,C PØINTS TØ BEG.ØF ARG TAB.
007133/ 315 CALL      006012   SWEEP ARG.TAB.
007136/ 013 DCX   B            LØAD ACC WITH LETTER IN ARG.T.
007137/ 012 LDAX  B            -
007140/ 301 PØP   B            RESTØRE B,C
007141/ 303 JMP       007055   PRINT CØNT ØF ACC
007144/ 176 MØV   A M          1 ARG DEST.
007145/ 346 NDI       307      RST?
007147/ 376 CPI       307      -
007151/ 302 JNZ       007176   NØ
007154/ 076 MVI   A   040      YES,FINISH ARG.FIELD
007156/ 006 MVI   B   004      WITH BLANC
007160/ 315 CALL      003263   -
007163/ 176 MØV   A M          INSTR INRØ ACC
007164/ 346 NDI       070      CØNVERT  TØ ØCT NØ.AND
007166/ 041 LXI   H   010027   SEND TØ MEM.
007171/ 167 MØV   M A          -
007172/ 315 CALL      001326   PRINT ØCTAL BYTE                ?
007175/ 311 RET
007176/ 176 MØV   A M          SP TEST
007177/ 346 NDI       365
007201/ 376 CPI       061      -
007203/ 312 JZ        007255   PRINT SP
007206/ 346 NDI       361      PSW TEST
007210/ 376 CPI       361
007212/ 312 JZ        007270   PRINT PSW
007215/ 116 MØV   C M          EXCEPTIØN TEST
007216/ 072 LDA       010011
007221/ 376 CPI       367      -
007223/ 302 JNZ       007230   NØ EXCEP.
007226/ 241 NDA   C            YES EXEPT.,DECR.ARG.CØDE BY 1
007227/ 117 MØV   C A          -
007230/ 076 MVI   A   070      GENERATING ØF
007232/ 241 NDA   C            ARG.CØDE
007233/ 017 RRC
007234/ 017 RRC
007235/ 017 RRC
007236/ 366 ØRI       370      -
007240/ 303 JMP       007127   GØ TØ PRINT
007243/ 315 CALL      007176   DEST.          2 ARG.PRINT
007246/ 315 CALL      001313   PRINT BLANC
007251/ 005 DCR   B            FILD CØUNT
007252/ 303 JMP       007124   SØURCE
007255/ 076 MVI   A   123                    PRINT SP
007257/ 315 CALL      001133
007262/ 076 MVI   A   120
007264/ 315 CALL      001133                                   ?
007267/ 311 RET                -
007270/ 076 MVI   A   120                    PRINT PSW
007272/ 315 CALL      001133
007275/ 076 MVI   A   123
```

```
007277/ 315 CALL      001133
007302/ 076 MVI   A   127
007304/ 315 CALL      001133
007307/ 311 RET                    -
007310/ 306 ADI       305          EXCEPT.            EXCEPT.
007312/ 312 JZ        007325       SUSPECT            TEST
007315/ 074 INR   A
007316/ 312 JZ        007325
007321/ 074 INR   A                -
007322/ 302 JNZ       007350       NØ EXEPT.,HLT TEST
007325/ 176 MØV   A M              NØ EXCEPT.
007326/ 376 CPI       072
007330/ 310 RZ                     -
007331/ 346 NDI       010          NØ EXC.
007333/ 310 RZ                     -
007334/ 076 MVI   A   367          YES EXC.
007336/ 062 STA       010011       SET EXC.FLAG
007341/ 072 LDA       010030       INCR.D,E TØ
007344/ 203 ADD   E                NEXT INSTR.IN TABLE
007345/ 137 MØV   E A
007346/ 023 INX   D
007347/ 311 RET                    -
007350/ 176 MØV   A M              HLT TEST
007351/ 376 CPI       166
007353/ 300 RNZ                    -
007354/ 227 SUB   A                HLT,SET ARG.FLAG
007355/ 062 STA       010027       AND INCR.D,E TØ NEXT INSTR.
007360/ 303 JMP       007341       -
020000/ 315 CALL      002227       FØRMAT          CPY RØUT.
020003/ 072 LDA       010016       PRØGRAMMER?
020006/ 376 CPI       001          -
020010/ 302 JNZ       020051       NØ
020013/ 072 LDA       010000       YES;L.LIMIT TØ ACC
020016/ 127 MØV   D A              L.LIM.TØ D
020017/ 072 LDA       010002       U.LIMIT TØ ACC
020022/ 222 SUB   D                U.L-L.L.TØ ACC
020023/ 117 MØV   C A              (U.L.-L.L.)TØ C
020024/ 052 LHLD      010004       NEW RANGE TØ H,L
020027/ 172 MØV   A D              L.L TØ ACC
020030/ 323 ØUT       020          DATA TØ ACC
020032/ 333 IN        002          -
020034/ 167 MØV   M A              DATA TØ MEM.
020035/ 043 INX   H                H,L = H,L PLUS 1
020036/ 024 INR   D                L.L. = L.L. PLUS 1
020037/ 015 DCR   C                CØUNT
020040/ 076 MVI   A   377          TEST
020042/ 271 CMP   C                -
020043/ 312 JZ        000000       FINISHED
020046/ 303 JMP       020027       REPEAT
020051/ 052 LHLD      010004       NØ PRG.,NEW RANGE TØ H,L
020054/ 072 LDA       010000       (L.L.-N.R.)TØ B,C
020057/ 225 SUB   L
020060/ 117 MØV   C A
020061/ 072 LDA       010001
020064/ 234 SBB   H
020065/ 107 MØV   B A              -
020066/ 311 RET
020067/ 322 JNC       020104       (L.L.-N.R.)>0?
020072/ 171 MØV   A C              MØVES UP;TWØS CØMPL.ØF(L.L.-N.R)
020073/ 057 CMA                    INTØ B,C
020074/ 117 MØV   C A
020075/ 170 MØV   A B
020076/ 057 CMA
020077/ 107 MØV   B A
020100/ 003 INX   B                -
020101/ 303 JMP       020146
020104/ 041 LXI   H   010000       MØVES DØWN
020107/ 176 MØV   A M              SUBTRACT (L.L.-N.R.)FRØM L.L.
020110/ 221 SUB   C
020111/ 137 MØV   E A
020112/ 054 INR   L
```

?

13 bytes spare

```
020113/ 176 MØV   A M
020114/ 230 SBB   B
020115/ 127 MØV   D A                 -
020116/ 052 LHLD      010000          CØPY DATA
020121/ 176 MØV   A M                 TØ NEW LØC.
020122/ 022 STAX  D
020123/ 043 INX   H
020124/ 023 INX   D
020125/ 072 LDA       010002
020130/ 275 CMP   L
020131/ 302 JNZ       020121
020134/ 176 MØV   A M
020135/ 022 STAX  D
020136/ 072 LDA       010003
020141/ 274 CMP   H
020142/ 302 JNZ       020121
020145/ 311 RET                       -
020146/ 041 LXI   H   010002          MØVES UP;(L.L.-N.R.)PLUS
020151/ 176 MØV   A M                 U.L. INTØ D,E
020152/ 201 ADD   C
020153/ 137 MØV   E A
020154/ 054 INR   L
020155/ 176 MØV   A M
020156/ 210 ADC   B
020157/ 127 MØV   D A                 -
020160/ 052 LHLD      010002          CØPY DATA TØ NEW LØC.
020163/ 176 MØV   A M
020164/ 022 STAX  D
020165/ 053 DCX   H
020166/ 033 DCX   D
020167/ 072 LDA       010000
020172/ 275 CMP   L
020173/ 302 JNZ       020163
020176/ 176 MØV   A M
020177/ 022 STAX  D
020200/ 072 LDA       010001
020203/ 274 CMP   H
020204/ 302 JNZ       020163
020207/ 311 RET                       -
020210/ 315 CALL      020000          CPY RØUT.STARTS HERE
020213/ 315 CALL      020067
020216/ 307 RST       000             -
------
020217/ 042 052 062 072 071           TABLE CØNTAINING CØDES FØR
------                                SHLD LHLD STA LDA LXI

020224/ 052 LHLD      010000          L.L.INTØ H,L          TRN RØUT.
020227/ 315 CALL      020246          CALCULATIØN
020232/ 370 RM                        -
020233/ 052 LHLD      010002          U.L.INTØ H,L
020236/ 315 CALL      020246          CALCULATIØN
020241/ 360 RP                        -
020242/ 341 PØP   H                   MØVE UP STACK
020243/ 303 JMP       021024          CHANGE THE ADDRESS
020246/ 353 XCHG                      CALCULAT.;SUBTRACT L.L.,U.L.,
020247/ 052 LHLD      010023          FRØM JMP ADDRESS
020252/ 043 INX   H
020253/ 176 MØV   A M
020254/ 223 SUB   E
020255/ 043 INX   H
020256/ 176 MØV   A M
020257/ 232 SBB   D
020260/ 311 RET                       -
020261/ 052 LHLD      010012          MMM PLUS (U.L.-L.L.)    END ØF
020264/ 353 XCHG                      INTØ D,E                PRØG?
020265/ 052 LHLD      010023          PØINTER INTØ H,L
020270/ 175 MØV   A L                 PØINTER-/MMM-(U.L.-L.L.)/>0?
020271/ 223 SUB   E
020272/ 174 MØV   A H
020273/ 232 SBB   D
020274/ 311 RET                       -
```

```
020275/ 315 CALL       002221       TRN RØUT STARTS HERE
020300/ 315 CALL       020051       UP ØR DØWN?
020303/ 322 JNC        020324       MØVES UP;
020306/ 171 MØV    A C              TWØS CØMPL.ØF L.L.-N.R
020307/ 057 CMA
020310/ 117 MØV    C A
020311/ 170 MØV    A B
020312/ 057 CMA
020313/ 107 MØV    B A
020314/ 003 INX    B                -
020315/ 227 SUB    A                MØVES UP FLAG
020316/ 062 STA        010014       -
020321/ 303 JMP        020331
020324/ 076 MVI    A   001           MØVES DØWN
020326/ 062 STA        010014       -
020331/ 315 CALL       021103       CALCULATE MMM PLUS (U.L.-L.L.)
020334/ 042 SHLD       010023       PØINTER INTØ H,L
020337/ 305 PUSH   B                SAVE B
020340/ 315 CALL       006230       MATCH RØUT.
020343/ 315 CALL       007310       EXCEPTIØN RØUT.
020346/ 301 PØP    B                RESTØRE B
020347/ 032 LDAX   D                CØDE TØ 010016
020350/ 062 STA        010016       -
020353/ 315 CALL       006377       FIRST LETTER SUBR.
020356/ 034 INR    E                SECØUND LETTER
020357/ 032 LDAX   D                DATA BYTE TEST
020360/ 036 MVI    E   300
020362/ 243 NDA    E                -
020363/ 376 CPI        000          NØ DATA BYTE
020365/ 312 JZ         021100       -
020370/ 376 CPI        100          1 DATA BYTE
020372/ 312 JZ         021077       -
020375/ 072 LDA        010016       ASSUME 2 DATA BYTES
021000/ 041 LXI    H   020217       IS IT SHLD,LHLD,STA,LDA,LXI?
021003/ 036 MVI    E   005
021005/ 276 CMP    M                -
021006/ 312 JZ         021073       YES,NØ CHANGE
021011/ 054 INR    L
021012/ 035 DCR    E
021013/ 302 JNZ        021005
021016/ 315 CALL       020224       ØUT ØF RANGE?
021021/ 303 JMP        021073       YES,NØ CHANGE
021024/ 052 LHLD       010023       CHANGE
021027/ 043 INX    H                -
021030/ 072 LDA        010014       MØVES DØWN?
021033/ 247 NDA    A
021034/ 302 JNZ        021061       -
021037/ 176 MØV    A M              MØVES UP;ADD (L.L.-N.R.)
021040/ 201 ADD    C                TØ ADDR.ØF JUMP
021041/ 167 MØV    M A
021042/ 043 INX    H
021043/ 176 MØV    A M
021044/ 210 ADC    B
021045/ 167 MØV    M A
021046/ 043 INX    H                -
021047/ 042 SHLD       010023       INCREMENTED PØINTER TØ 010023
021052/ 315 CALL       020261       END ØF PRØG?
021055/ 372 JM         020334       NØ,NEW LINE
021060/ 307 RST        000          YES
021061/ 176 MØV    A M              MØVES DØWN;SUBTRACT
021062/ 221 SUB    C                (L.L.-N.R.)FRØM ADDR.
021063/ 167 MØV    M A
021064/ 043 INX    H
021065/ 176 MØV    A M
021066/ 230 SBB    B
021067/ 167 MØV    M A              -
021070/ 303 JMP        021046       NEW LINE
021073/ 052 LHLD       010023       NØ CHANGE
021076/ 043 INX    H
021077/ 043 INX    H
021100/ 303 JMP        021046       -
```

```
021103/ 305 PUSH  B                   SAVE B,C
021104/ 052 LHLD      010002          U.L.-L.L.
021107/ 353 XCHG
021110/ 052 LHLD      010000
021113/ 173 MØV   A E
021114/ 225 SUB   L
021115/ 137 MØV   E A
021116/ 172 MØV   A D
021117/ 234 SBB   H
021120/ 127 MØV   D A                 -
021121/ 052 LHLD      010006          MMM PLUS (U.L.-L.L.)
021124/ 173 MØV   A E                 INTØ 010012
021125/ 205 ADD   L
021126/ 157 MØV   L A
021127/ 172 MØV   A D
021130/ 214 ADC   H
021131/ 147 MØV   H A
021132/ 042 SHLD      010012          -
021135/ 052 LHLD      010006
021140/ 301 PØP   B                   RESTØRE B,C
021141/ 311 RET
021142/ 315 CALL      002235          I/P L.L.,U.L.       PRG.RØUTIN
021145/ 315 CALL      001355          CR/LF
021150/ 076 MVI   A   045             Ø/P %
021152/ 315 CALL      001133          -
021155/ 315 CALL      001001          I/P LETTER
021160/ 007 RLC                       SHIFT,STØRE
021161/ 007 RLC
021162/ 137 MØV   E A                 -
021163/ 052 LHLD      010000          L.L.INTØ H,L
021166/ 175 MØV   A L                 CØMPARE RØM WITH MEM.
021167/ 323 ØUT       020
021171/ 333 IN        002
021173/ 276 CMP   M                   -
021174/ 304 CNZ       021225          PRØGR.IF NØT EQUAL
021177/ 072 LDA       010002          END ØF PRØGRAMM?
021202/ 275 CMP   L
021203/ 302 JNZ       021221
021206/ 072 LDA       010003
021211/ 274 CMP   H
021212/ 302 JNZ       021221          -
021215/ 315 CALL      021225          YES,PRØGR.LAST
021220/ 307 RST       000             -
021221/ 043 INX   H                   NEW BYTE
021222/ 303 JMP       021321          -
021225/ 006 MVI   B   001             PRØGR.4 TIMES LØNGER    PRØGRAMMING
021227/ 315 CALL      021252
021232/ 170 MØV   A B
021233/ 007 RLC
021234/ 007 RLC
021235/ 107 MØV   B A
021236/ 315 CALL      021252
021241/ 005 DCR   B
021242/ 302 JNZ       021236          -
021245/ 170 MØV   A B                 Ø/P NULL CHARACT.
021246/ 315 CALL      001133
021251/ 311 RET                       -
021252/ 176 MØV   A M                 PRØGRAM UNTILL RØM
021253/ 057 CMA                       BYTE HØLDS DESIRED
021254/ 323 ØUT       022             DATA;MAX 255 PRG.PULSES
021256/ 076 MVI   A   004
021260/ 323 ØUT       026
021262/ 257 XRA   A
021263/ 323 ØUT       026
021265/ 113 MØV   C E
021266/ 026 MVI   D   350
021270/ 025 DCR   D
021271/ 302 JNZ       021270
021274/ 015 DCR   C
021275/ 302 JNZ       021266
021300/ 333 IN        002
```

```
021302/ 276 CMP   M
021303/ 310 RZ
021304/ 004 INR   B
021305/ 302 JNZ       021252      -
021310/ 315 CALL      002025      ERRØR;Ø/P ADDRESS,?
021313/ 076 MVI   A   077
021315/ 315 CALL      001133
021320/ 307 RST         000       -
021321/ 042 SHLD        010000
021324/ 303 JMP         021166
```

41 bytes spare.